

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

**IMPLEMENTACE KOMUNIKAČNÍCH TECHNOLOGIÍ IOT V
SIMULAČNÍM NÁSTROJI NS-3**

IMPLEMENTATION OF IOT COMMUNICATION TECHNOLOGIES IN NS-3 SIMULATION TOOL

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Marek Rehberger

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Radovan Juráň

BRNO 2021

Bakalářská práce

bakalářský studijní program **Telekomunikační a informační systémy**

Ústav telekomunikací

Student: Marek Rehberger

ID: 214089

Ročník: 3

Akademický rok: 2020/21

NÁZEV TÉMATU:

Implementace komunikačních technologií IoT v simulačním nástroji NS-3

POKYNY PRO VYPRACOVÁNÍ:

Bakalářská práce má za cíl ve své teoretické části prostudovat a porovnat dvě perspektivní komunikační technologie LPWA v licenčních pásmech, tj. LTE Cat-M1 a NB-IoT. Následně bude v praktické části pro jednu z uvedených technologií provedena definice a implementace komunikačního scénáře pro přenos dat v rámci chytrých sítí (SmartGrids) s využitím nástroje Network Simulator 3 (NS-3) a modulu LENA / LENA 5G.

DOPORUČENÁ LITERATURA:

[1] Network Simulator 3: Documentation, A Discrete-Event Network Simulator [online], 2020. Dostupné z: <https://www.nsnam.org/doxygen>

[2] LIBERG, Olof, Marten SUNDBERG, Y.P. Eric WANG, Johan BERGMA a Joachim SACHS, [2018]. Cellular Internet of things: technologies, standards, and performance. San Diego, CA, United States: Academic PRes, an imprint of Elsevier: ISBN 978-012-8124-581.

Termín zadání: 1.2.2021

Termín odevzdání: 31.5.2021

Vedoucí práce: Ing. Radovan Juráň

prof. Ing. Jiří Mišurec, CSc.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Bakalářská práce se ve své teoretické části zabývá LPWA komunikačními technologiemi v licenčním frekvenčním spektru, konkrétně technologiemi LTE Cat-M1 a NB-IoT. V teoretické části jsou shrnuty pojmy týkající se LPWAN (Low Power Wide Area Network), licenční a bezlicenční technologie, chytré sítě, NB-IoT a LTE Cat-M1. Je popsáno využití LPWA technologií v oblasti chytrých sítí a pro práci je využita myšlenka chytrých odečtů v budovách. Následně je v praktické části pro technologii LTE Cat-M1 provedena implementace komunikačních scénářů pro přenos dat v rámci chytrých sítí s využitím nástroje Network Simulator 3 (NS-3) a modulu LENA. Výsledky simulací jsou přehledně zpracovány formou grafů a tabulek.

KLÍČOVÁ SLOVA

IoT (Internet věcí), Chytrá síť, Chytré odečty, LPWAN, NB-IoT (Úzkopásmové IoT), LTE Cat-M1, LTE, NS-3, LENA

ABSTRACT

The bachelor thesis deals in its theoretical part with LPWA communication technologies in the licensed frequency spectrum, specific technologies LTE Cat-M1 and NB-IoT. The theoretical part summarizes the terms concerning to LPWAN (Low Power Wide Area Network), licensed and unlicensed technologies, SmartGrids, NB-IoT and LTE Cat-M1. The use of LPWA technologies in the field of SmartGrids is described and the idea of Smart Metering in buildings is used in the work. Subsequently, in the practical part for the LTE Cat-M1 technology, the implementation of communication scenarios for data transmission within SmartGrids is performed using the tools Network Simulator 3 (NS-3) and the LENA module. The results of the simulations are clearly processed in the form of graphs and tables.

KEYWORDS

IoT (Internet of Things), SmartGrids, Smart metering, LPWAN, NB-IoT (Narrowband IoT), LTE Cat-M1, LTE, NS-3, LENA

REHBERGER, Marek. *Implementace IoT komunikačních technologií v simulačním nástroji NS-3*. Brno, 2021, 68 s. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Radovan Juraň

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Implementace IoT komunikačních technologií v simulačním nástroji NS-3“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Radovanovi Juraňovi, za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

Úvod	11
1 LPWAN a SmartGrids	12
1.1 Definice LPWAN	12
1.1.1 Bezlicenční technologie	13
1.1.2 Licenční technologie	14
1.2 Definice SmartGrids	15
1.3 LPWAN ve SmartGrids	15
2 Technologie NB-IoT	17
2.1 Úvod do technologie	17
2.2 Provozní využití	17
2.2.1 Standalone provoz	18
2.2.2 In-Band provoz	18
2.2.3 Guard Band provoz	18
2.3 Systém model	18
2.3.1 NPRACH	19
2.3.2 NPUSCH formát 1 (Data)	20
2.3.3 NPUSCH formát 2 (Řízení)	21
2.4 Technická specifikace	21
2.4.1 Vysílací výkon	21
2.4.2 Frekvence	22
2.4.3 Teoretický dosah	22
2.4.4 Pokrytí	23
2.4.5 Datová rychlost	23
2.4.6 Souhrn technických parametrů	23
3 Technologie LTE Cat-M1	24
3.1 Úvod do technologie	24
3.1.1 Architektury sítě LTE	24
3.2 Systém model	25
3.2.1 Scénář 1: Vyšší priorita provozu IoT	27
3.2.2 Scénář 2: Vyšší priorita LTE provozu	27
3.2.3 Scénář 3: Provoz IoT i provoz LTE mají stejnou prioritu	27
3.3 Pokrytí	27
3.3.1 Provozní režimy pro vylepšené pokrytí	28
3.4 Technická specifikace	29

3.4.1	Vysílací výkon	29
3.4.2	Frekvence	29
3.4.3	Teoretický dosah	29
3.4.4	Datová rychlost	30
3.4.5	Souhrn technických parametrů	30
4	Srovnání NB-IoT a LTE Cat-M1	31
4.1	Spotřeba	31
4.2	Dostupnost služeb v globálním kontextu	31
4.3	Rozsah pokrytí	33
4.4	Připojení	33
4.5	Náklady	33
4.6	Shrnutí	33
5	Network Simulator 3	35
5.1	Úvod do NS-3	35
5.2	Modul LENA	35
5.3	Možné výsledky z simulací	36
6	Scénáře z NS-3	37
6.1	Počáteční scénář NS-3	37
6.1.1	Výpis kódu	38
6.2	Rozšířený scénář NS-3	42
6.2.1	Výpis kódu	43
7	Výsledky simulací pro rozšířený scénář	55
7.1	Scénář se 100 UEs	55
7.2	Scénář s 200 UEs	57
7.3	Scénář s 250 UEs	60
7.4	Přehled výsledků v tabulce	62
	Závěr	64
	Literatura	65
A	Obsah zdrojového souboru	68

Seznam obrázků

1.1	Teoretický dosah různých IoT bezdrátových technologií, převzato z [20], upraveno	13
1.2	Obecná architektura pro systém inteligentní sítě používaný v inteligentním městě (Smart City), převzato z [8], upraveno	15
2.1	NB-IoT provozní režimy, převzato z [1]	17
2.2	Struktura NPUSCH a NPRACH, převzato z [9]	19
2.3	Struktura času a frekvence NPRACH, převzato z [2]	20
2.4	Struktura zdrojových jednotek NPUSCH Format 1 (data), převzato z [2]	21
2.5	Datová rychlost a komunikační rozsah pro komunikační sítě, převzato z [11], upraveno	22
3.1	Síťová architektura LTE, převzato z [21], upraveno	24
3.2	Systém model LTE Cat-M1, převzato z [24], upraveno	25
3.3	Vylepšení cílového pokrytí LTE Cat-M1, převzato z [7], upraveno	28
4.1	Mapa zastoupení technologie NB-IoT a LTE Cat-M1 ve světě 2020, převzato z [6]	32
6.1	Topologie sítě počátečního scénáře v programu NetAnim	37
6.2	Topologie sítě rozšířeného scénáře v programu NetAnim	43
7.1	První scénář – Přenosová rychlost	56
7.2	První scénář – Zpoždění	56
7.3	První scénář – Ztrátovost paketů	57
7.4	Druhý scénář – Přenosová rychlost	58
7.5	Druhý scénář – Zpoždění	59
7.6	Druhý scénář – Ztrátovost paketů	59
7.7	Třetí scénář – Přenosová rychlost	60
7.8	Třetí scénář – Zpoždění	61
7.9	Třetí scénář – Ztrátovost paketů	62

Seznam tabulek

1.1	Srovnání vybraných zástupců LPWA technologií v bezlicenčních ISM pásmech, převzato z [20]	14
1.2	Tabulka informací o technologii EC-GSM-IoT, převzato z [20]	14
2.1	Souhrn technických parametrů NB-IoT, převzato z [20]	23
3.1	Souhrn technických parametrů, převzato z [20]	30
4.1	Výpis operátorů 2020 (umístění a použité technologie), převzato z [5]	32
6.1	Pojmy RX/TX bytes a RX/TX packets	51
7.1	Přehled parametrů pro scénář se 100 UEs	55
7.2	Přehled parametrů pro scénář s 200 UEs	58
7.3	Přehled parametrů pro scénář s 250 UEs	60
7.4	Přehled výsledků pro uplink	62

Seznam výpisů

6.1	Definice proměnných na začátku	38
6.2	Vytvoření uzlů v síti	38
6.3	Vytvoření internetu	39
6.4	Instalace mobilního modulu	39
6.5	Instalace LTE zařízení na uzly	40
6.6	Přiřazení jednoho UE k jednomu eNodeB	40
6.7	Instalace a start aplikací	40
6.8	PCAP trasování	41
6.9	Animace v NetAnim	41
6.10	Implementace FlowMonitor	41
6.11	Obsah souboru FlowMonitor.xml	42
6.12	Definice proměnných na začátku	43
6.13	Definice argumentů pro příkazový řádek	44
6.14	Příkazový řádek	44
6.15	Konfigurace vysílacích parametrů	44
6.16	Konfigurace SRS	45
6.17	Výpis do konzole o délce simulace	45
6.18	Vytvoření uzlů PGW a Remote_Host	45
6.19	Nastavení internetu část 1 (linka)	46
6.20	Nastavení internetu část 2 (IP adresy)	46
6.21	Přiřazení IP adres k UEs	46
6.22	Vytvoření UE_n a eNodeB_n	47
6.23	Mobility model	47
6.24	Připojení UE na eNodeB v aktuálním čase	47
6.25	Vytvoření BulkSendAplikace	48
6.26	Vytvoření PacketSinkAplikace	49
6.27	Definice animace část 1	49
6.28	Definice UEs a animace část 2	50
6.29	Nástroj FlowMonitor	50
6.30	Nástroj Gnuplot	51
6.31	Výpis parametrů do konzole a souboru	52

Úvod

Práce se věnuje LPWAN sítím v licenčních pásmech. Na začátku je proveden popis LPWAN bezlicenčních a licenčních technologií spolu s pojmem SmartGrids. Konkrétně se práce hlavně zaměřuje na dvě perspektivní M2M komunikační technologie – LTE Cat M1 a NB-IoT v aplikacích z oblasti SmartGrids. Pro obě sítě jsou srovnány jejich technické parametry z hlediska šířky pásma, životnosti baterie a nebo teoretického dosahu.

LPWAN sítě pojmu velké množství zařízení, a tak nacházejí uplatnění mimo jiné i v aplikacích tzv. inteligentních sítí (SmartGrids), kde jsou schopny pokrýt požadavky na bezdrátovou konektivitu a značný počet připojených zařízení. SmartGrids jsou velkou příležitostí k rozvoji infrastruktury měst a různých hi-tech řešení.

Teoretická část zahrnuje vybrané bezdrátové komunikační technologie, které do této kategorie spadají. Jsou jimi technologie LTE Cat-M1 a NB-IoT, které pracují v licenčním pásmu. Toto pásmo je placené, ale poskytuje lepší vlastnosti připojení. Obě technologie jsou srovnány z hlediska efektivity v různých oblastech využití. Dále jsou definovány jejich přednosti vzhledem k technickým parametrům, které ovlivňují jejich samotné použití v terénu. Technické parametry jsou definovány v těchto bodech jako je vysílací výkon, frekvence, teoretický dosah, pokrytí a nebo datová rychlost.

Praktická část se zabývá už pouze jednou zvolenou technologií a tím je LTE Cat-M1, která je simulována v simulátoru NS-3 za využití programu NetAnim. Scénář je psán v programu C++ a převáděn do simulace NetAnim. Na konci je definován celkový rozbor simulace a změřených výsledků. Dále je proveden popis simulace a přiložených výstupů.

1 LPWAN a SmartGrids

1.1 Definice LPWAN

LPWAN (Low Power Wide Area Network) jsou obecně nízkopříkonové komunikační technologie, které umožňují bezdrátový přenos na velké vzdálenosti s minimální energetickou spotřebou a nízkou údržbou. Srovnání s ostatními soudobými technologiemi je na obrázku 1.1. LPWAN nejlépe vystihuje právě energetickou efektivitu, škálovatelnost a pokrytí relativně velkých oblastí. Poskytuje praktický a ekonomický způsob pro vytváření sítí IoT [23].

LPWAN sítě jsou používány pro M2M komunikaci, které se provozují jak v bezlicenčním, tak i v licenčním pásmu. V licenčním pásmu existují technologie, které používají stávající infrastrukturu původně určenou pro H2H komunikaci. Tyto technologie mají odlišné požadavky na datové rychlosti a objemy dat než má původní infrastruktura.

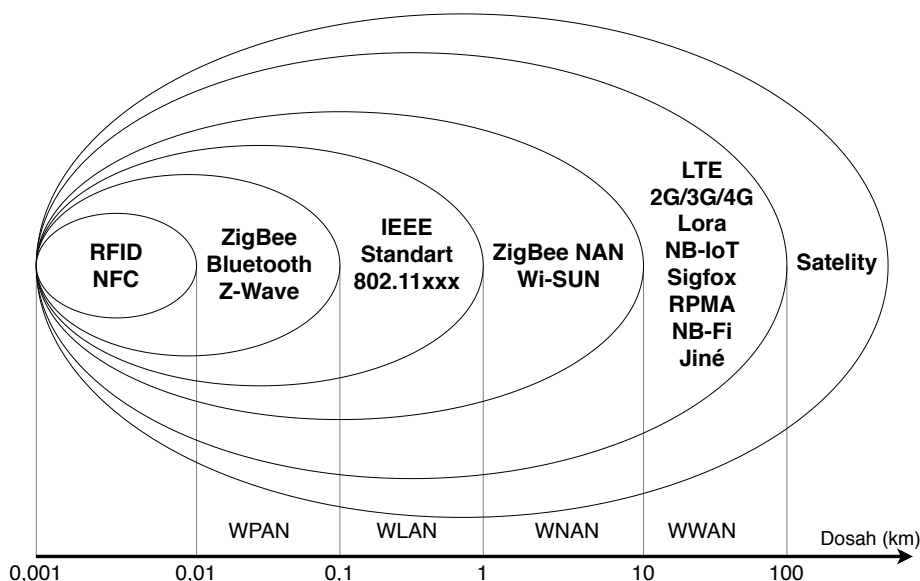
H2H (Human-to-Human) komunikace klade požadavky na vysoké datové rychlosti za účelem poskytování sofistikovanějších služeb, jako jsou hovory a streamování videa, a vede k vývoji složitějších zařízení. Zatímco M2M (Machine-to-Machine) komunikace má opačné cíle – nízká přenosová rychlost s následným omezeným využitím šířky pásma a velmi nízká spotřeba energie za předpokladu výdrže baterie minimálně 10 let [18].

M2M komunikace, s ohledem na původní buňkové sítě, přispěla ke vzniku technologií za účelem vývoje specializované infrastruktury pro IoT (Internet of Things). Pro tento typ se využívají stávající infrastruktury jako jsou GSM nebo LTE, které poskytují předvídatelné a kontrolovatelné prostředí v hustě zarušeném sektoru zařízeními. M2M požadavky na nízko nákladová zařízení, všudypřítomné pokrytí a ultra nízkou spotřebu se neustále vyvíjí. Tento typ komunikace je řešen a implementován také do technologií LTE-M a NB-IoT [18].

V dnešní době začíná růst význam těchto technologií prakticky ve všech oblastech průmyslu i v běžném životě [20]. Nacházejí uplatnění například v oblasti zemědělství pro inteligentní zavlažovací systémy, v domácnostech pro inteligentní správu energie nebo také v dopravě. Podrobnější informace najdeme v článku [8].

Jedním z cílů technologií LPWA je návrh inteligentních zařízení, které budou spotřebovávat méně energie a tím se prodlouží jejich životnost, zejména při provozu na baterie. Překážkou může být nedostatek vhodných komunikačních kanálů, proto v některých případech je zapotřebí jiných technologií k navázání spojení mezi různými platformami [23].

Mezi technologie, které reprezentují LPWAN, řadíme například NB-IoT (Narrow-Band Internet of Things) a LoRa (Long Range). LoRa technologie je dále obohacena



Obr. 1.1: Teoretický dosah různých IoT bezdrátových technologií, převzato z [20], upraveno

o komunikační protokol LoRaWAN. Další technologie jsou např. SigFox, ale také zde patří LTE-M a mnoho dalších technologií. Tyto technologie můžeme dělit na licenční a bezlicenční [15]. V současné době neexistuje jediná technologie ani v licenčním ani bezlicenčním spektru, která by byla jedinou tzv. killer app. To si můžeme vysvětlit jako počítačový program, který je mnohem lepší než všechny ostatní svého druhu. To je jeden z důvodů, proč stojí za to vzájemně porovnávat a zjišťovat jejich silné a slabé stránky. Jedním z nástrojů pro srovnání jsou i simulace, které šetří náklady na skutečný hardware a měření v terénu.

1.1.1 Bezlicenční technologie

Do bezlicenčních technologií můžeme zařadit LoRa a SigFox. Mezi méně známé patří RPMA (Random Phase Multiple Access) a NB-Fi (Narrowband Fidelity). Podrobné specifikace jednotlivých technologií jsou v tabulce 1.1. Samotná zařízení vysílají v bezlicenčních frekvenčních pásmech. Jedná se tedy o volně přístupná pásma, které jsou bez licenčních poplatků [20].

Lze hovořit také o ISM (Industrial, Scientific and Medical) pásmu, které představují bezlicenční pásmo. Jedni z rozsahů z nich mohou být pásma např.

- 902 – 928 MHz,
- 2400 – 2500 MHz
- a 5,725 – 5,875 GHz.

ISM pásma jsou využívána komunikačními systémy jako Bluetooth, NFC, Wi-Fi a jiné [14].

Tab. 1.1: Srovnání vybraných zástupců LPWA technologií v bezlicenčních ISM pásmech, převzato z [20]

Technologie	LoRa	SigFox	RPMA	NB-Fi
Šířka pásma	125 kHz	100 Hz	1 MHz	100 Hz
Spektrum	ISM Sub GHz	ISM 900 MHz	ISM 2,4 GHz	ISM Sub GHz
Životnost baterie	10 let	10 let	10 let	10 let
Přenosová rychlost	0,29 – 50 kbps	100 bps	624 kbps	10 – 100 bps
Dvoucestná data	Half duplex	Half duplex	Half duplex	Full duplex
Zabezpečení	Uživatel	Uživatel	AES 128	XTEA 256
Škálovatelnost	Nízká	Nízká	Vysoká	Vysoká
Dosah (km)	15	10 – 30	5	17 – 50
Podpora	Ano	Ne	Ne	Ne
Nejlepší citlivost	–137 dB	–129 dB	–137 dB	–154 dB
Minimální náklady	5 \$	5 \$	20 \$	2 \$
TX spotřeba	22 mA @14 dB	45 mA @14 dB	245 mA @23 dB	44 mA @14 dB
Společnost	Lora Alliance	Sigfox	Ingenu	Waviot

1.1.2 Licenční technologie

Licenční technologie, kam patří např. NB-IoT, LTE-M nebo méně známá technologie EC-GSM-IoT, používají licenční frekvenční pásmo. Pásma jsou spravována mobilními operátory. Tento typ připojení je bezpečný a spolehlivý, protože využívá vyhrazený prostor v rámci frekvenčního spektra, tudíž nedochází k rušení mezi jednotlivými zařízeními. Disponuje možností vyšší rychlosti datového připojení [20]. Jedna z licenčních technologií je popsána v tabulce 1.2. Další z technologií jako NB-IoT a LTE Cat-M1 jsou podrobněji specifikovány dále v kapitole 2 a 3.

Tab. 1.2: Tabulka informací o technologii EC-GSM-IoT, převzato z [20]

Technologie	EC-GSM-IoT
Šířka pásma	200 kHz / kanál
Spektrum	In-Band GSM
Životnost baterie	5 let
Přenosová rychlost	70–240 kbps
Dvoucestná data	Full duplex
Zabezpečení	3GPP (128-256)
Škálovatelnost	Vysoká
Dosah (km)	15
Podpora	Ne
Nejlepší citlivost	Nespecifikováno
Minimální náklady	15 \$
TX spotřeba	Nespecifikováno
Společnost	3GPP

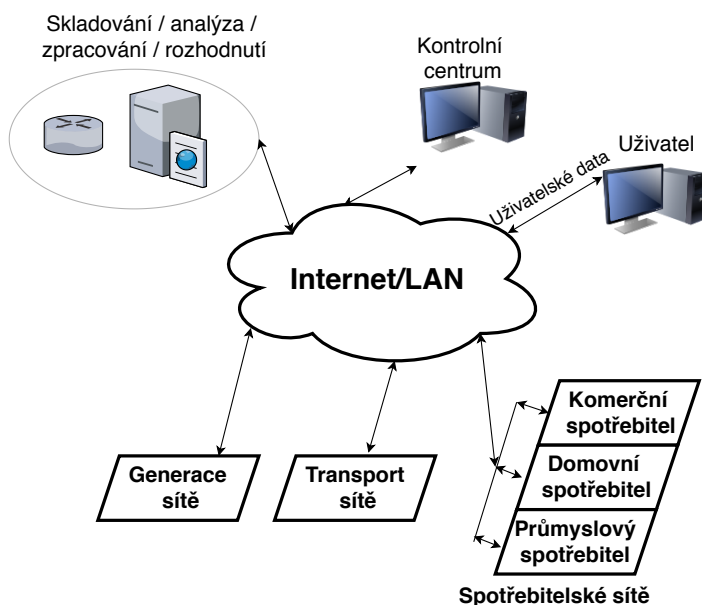
1.2 Definice SmartGrids

SmartGrid neboli inteligentní sítě jsou nový koncept elektrické distribuční soustavy, který integruje automatizované řízení a moderní komunikační technologie k vylepšení efektivity, spolehlivosti a bezpečnosti [11]. Zavedení CRSNs (Cognitive Radios Sensor Networks) a WSNs (Wireless Sensor Networks) do inteligentních sítí (Smart-Grid) vylepšuje možnosti výroby elektřiny, její přenos a distribuci ke koncovým zákazníkům. Energetické společnosti mohou monitorovat, předvídat, kontrolovat a řídit spotřebu energie efektivně a to s ohledem na provozní náklady [17].

Pomocí SmartGrid lze komunikovat s koncovými a vzdálenými zařízeními v reálném čase, a to za pomoci SmartGrid aplikací. Aplikací, jako například správa a řízení zátěže, bezdrátové automatické odečty měřičů, diagnostika a detekce poruch, dálkové ovládání a monitorování elektrického vedení a automatická distribuce [17].

1.3 LPWAN ve SmartGrids

LPWAN technologie mohou být aplikovány v mnoha oblastech např. jako inteligentní město (Smart City), inteligentní budovy (Smart Buildings), inteligentní domácnost (Smart Home), inteligentní odečty (Smart Metering) atd. Obecně SmartGrids jsou silové elektrické a komunikační sítě, které regulují výrobu a spotřebu elektrické energie [11]. Obecný příklad inteligentní sítě ve městě je na obrázku 1.2.



Obr. 1.2: Obecná architektura pro systém inteligentní sítě používaný v inteligentním městě (Smart City), převzato z [8], upraveno

Inteligentní síť obsahuje velké množství zařízení či senzorů, které samostatně hlásí své informace na konkrétní servery. LPWAN může poskytnout zcela nové a výhodné řešení pro uspokojení komunikačních požadavků inteligentní sítě [11].

Názorným příkladem využití chytrých čidel může být chytrý dům, vybavený teplotními senzory. Porovnáním vnitřní a vnější teploty lze pak adekvátně regulovat teplotu v jednotlivých místnostech dle požadavků uživatele. Vše bude zautomatizované s minimální účastí uživatele.

S pomocí technologií LPWAN je možné do značné míry zabránit poruchám zařízení a výpadkům dodávek elektrické energie. Využití LPWAN pro SmartGrid na bázi CRSN, naráží na určité limitace, jako například omezená životnost baterie, která má za následek nízkou schopnost zpracování dat. SmartGrid aplikace mají velké požadavky na kvalitu služeb (QoS – Quality of Service), které je náročné splnit pro QoS (nízké zpoždění nebo ztrátovost dat) a to kvůli různým charakteristikám síťových parametrů se vyžaduje použití integrovaného CRSN [17]. Pro bezlicenční technologie LPWAN je obtížné splnit tyto požadavky, protože je velmi pravděpodobné, že budou trpět rušením v přeplněném nelicencovaném pásmu, proto je vhodnější licencované pásmo s vyhrazenými prostředky. V těchto pásmech pracují i technologie NB-IoT a LTE Cat-M1, které jsou předmětem této práce [11].

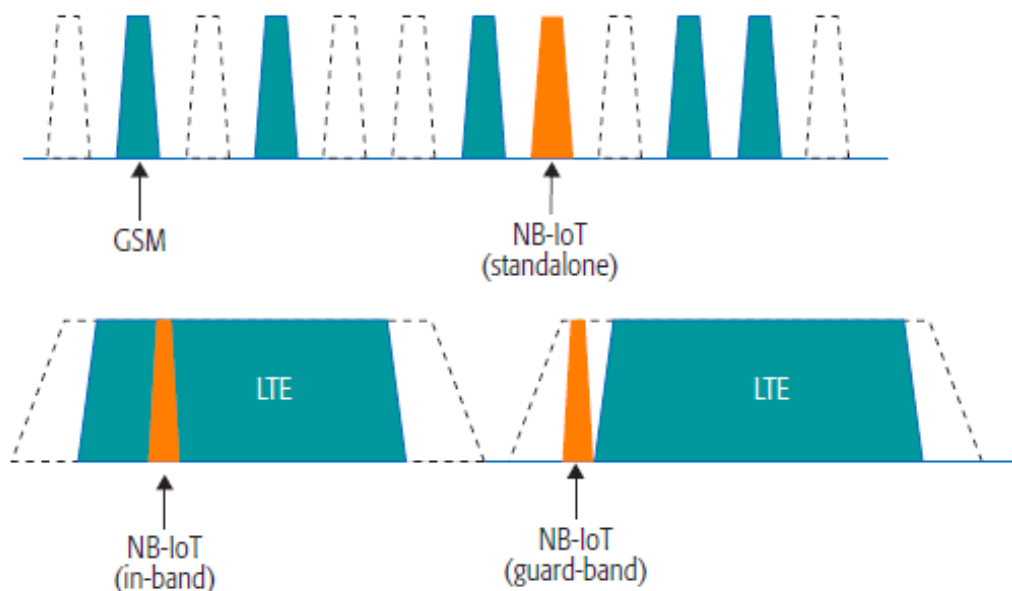
2 Technologie NB-IoT

2.1 Úvod do technologie

NB-IoT (NarrowBand-Internet of Things) je licencovaná rádiová technologie, vyvíjena společenstvím organizací a firem 3GPP (3rd Generation Partnership Project), která poskytuje spolehlivé připojení mezi levnými zařízeními IoT s rozšířeným pokrytím signálu a nízkou spotřebou energie. NB-IoT je postaveno na existující architektuře LTE (Long Term Evolution), proto využívá některé LTE funkce s přidánými rozšířeními, které používají modulaci OFDM (Orthogonal Frequency Division Multiplexing). Řada LTE funkcí je zjednodušena, optimalizována nebo rozšířena tak, aby byly splněny požadavky na nízkou rychlost dat, široké pokrytí a tak dále. Cílem vývoje je dosáhnout rychlé specifikace a minimalizovat vývojové úsilí [9, 20].

2.2 Provozní využití

Existují tři typy provozních režimů pro technologii NB-IoT: In-Band, Guard-Band a Standalone, tak jak je uvedeno na obrázku 2.1. NB-IoT má velmi dobré pokrytí, protože se spoléhá na stávající infrastrukturu 4G sítí, takže může dobře fungovat uvnitř budovy i v hustě zastavených oblastech. Také má dostatečnou dobu odezvy, což zvyšuje kvalitu služeb (QoS). K normálnímu provozu nepotřebuje vstupní bránu, využívá tzv. buňky, které mohou pokrývat až 50 000 koncových zařízení [20].



Obr. 2.1: NB-IoT provozní režimy, převzato z [1]

2.2.1 Standalone provoz

Ve Standalone provozu neboli samostatném provozu, NB-IoT by bylo typicky nasazeno v rámci jednoho nebo více stávajících GSM nosných. V samostatném provozu lze pro NB-IoT použít veškerý dostupný přenosový výkon stanice [1].

2.2.2 In-Band provoz

Nasazení In-Band provozu neboli provozu uvnitř pásma v rámci LTE nosné používá stejné PRBs (Pseudorandom binary sequences) jako LTE. V tomto režimu nasazení existují určitá omezení plánování využití prostředků, protože některé jsou vyhrazeny pro provoz LTE (např. pro fyzický downlink řídicí kanál a referenční signály). Vysílací výkon je sdílen mezi provozem LTE a NB-IoT [1].

2.2.3 Guard Band provoz

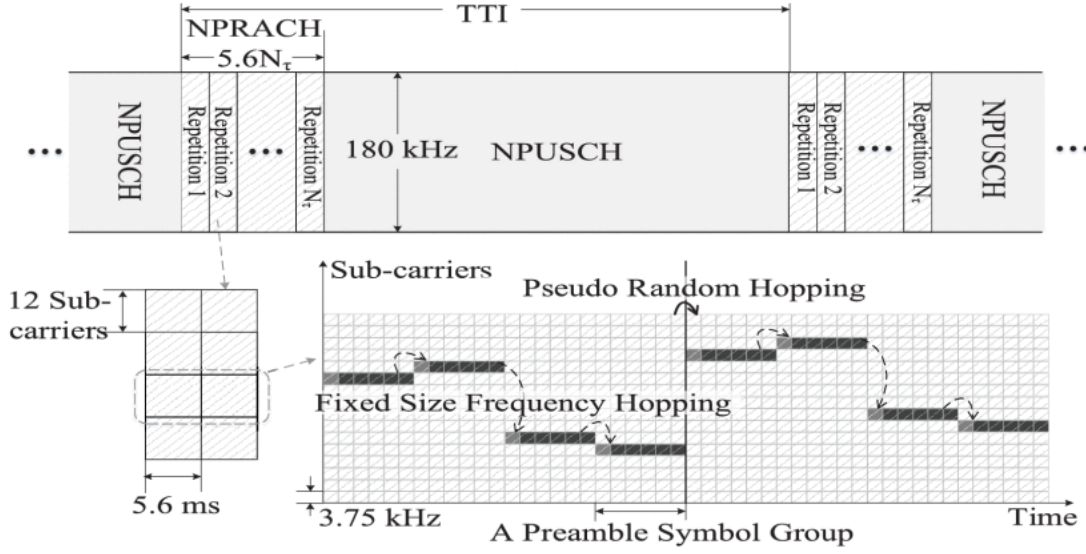
Využívají se nevyužité bloky zdrojů v Guard Band pásmu neboli ochranném intervalu LTE nosné a to z důvodu zajištění provozu obou kanálů, aby nedocházelo k rušení signálu. Obecně menší rušení je očekáváno ve srovnání s In-Band provozem, protože LTE rušení se nachází pouze na jedné straně nosiče NB-IoT. Tzv. buňka NB-IoT je obsluhována stejnou stanicí, rovněž sdílí maximální vysílací výkon stanice [1].

Jelikož je šířka pásma kanálu NB-IoT ve srovnání s LTE snížena, jsou fyzické signály a kanály přepracovány tak, aby vyhovovaly 180 kHz pro všechny potřebné požadavky. Pro zvýšenou kapacitu systému je možné provozovat NB-IoT v režimu multi-PRB, kde jedna nosná se používá k přenosu běžných signálů a kanálů. Jako jsou synchronizační signály a informace o vysílacím systému, a sekundární nosné [1].

2.3 Systém model

V uplinku NB-IoT se pro přenos dat používá NPUSCH (Narrowband Physical Uplink Shared Channel) a NPRACH (Narrowband Physical Random Access Channel) se používá pro přenos preamble. Šířka pásma 180 kHz je obsazena 3,75 kHz roztečí tónu (tj. zabírá přes 48 dílčích nosných) nebo rozteč tónu 15 kHz (tj. rozpětí přes 12 dílčích nosných), kde NPRACH podporuje pouze rozteč tónu 3,75 kHz. Pro NB-IoT síť lze nastavit až 3 hodnoty opakování z množiny (1, 2, 4, 8, 16, 32, 64, 128) v buňce, a umožňuje flexibilní konfiguraci zdroje NPRACH. V tomto modelu považujeme jednu opakující hodnotu za poskytnutí základních poznatků z důvodu opakování, kde přiřazení souvisejících zdrojů NPRACH probíhá pouze na začátku

časového intervalu přenosu (TTI – Transmission Time Interval) [9]. Jak je znázor-
něno na obrázku 2.2.



Obr. 2.2: Struktura NPUSCH a NPRACH, převzato z [9]

Vzhledem ke konfiguraci opakování jediné hodnoty bude pro každé aktivní IoT zařízení obsaženo na všech 48 subnosných, a tím každá subnosná má stejnou pravděpodobnost $\left(\frac{1}{48}\right)$ na to být vybrán [9].

Správná hodnota opakování může pomoci zlepšit efektivitu kanálu, protože nedostatečné opakování může vést k nízké pravděpodobnosti úspěchu RACH, ale redundantní opakování může vést k nedostatku NPUSCH. Poměr časové neobsazenosti (tj. poměr mezi dobou NPUSCH a TTI) je vyjádřen jako

$$T_a = (TTI - 5.6N_r)/TTI,$$

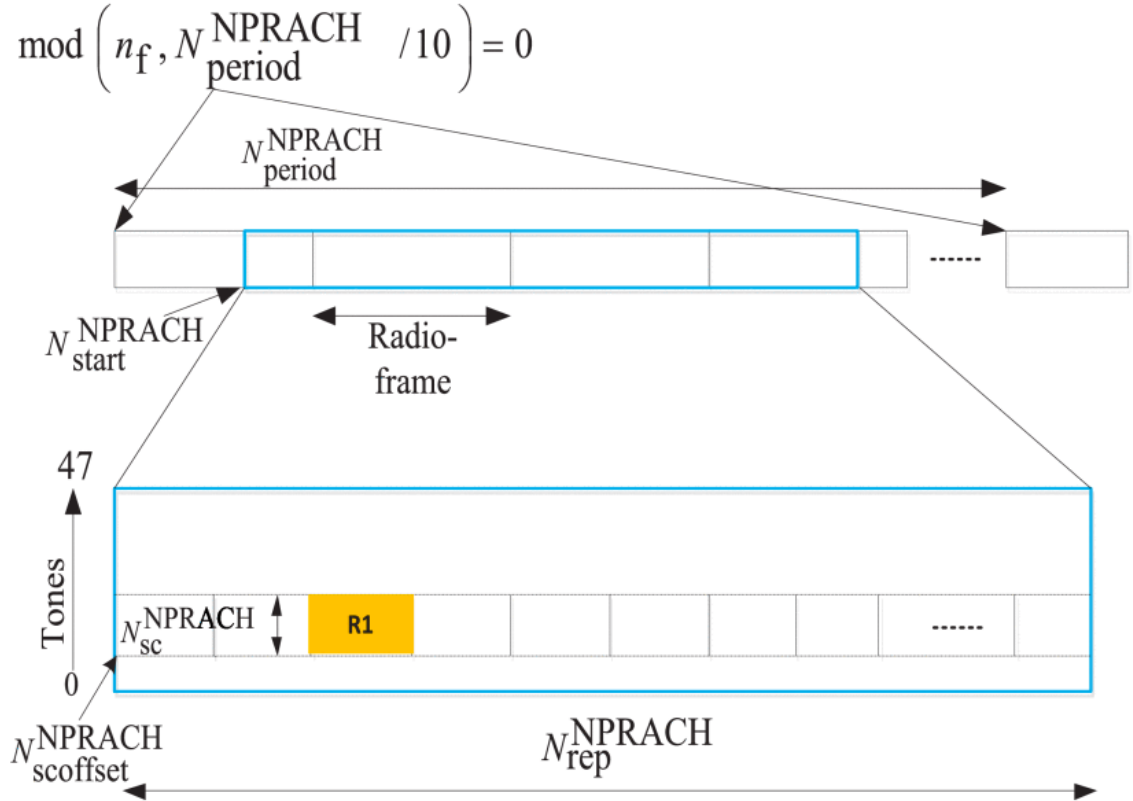
kde N_r je požadovaná hodnota opakování [9].

2.3.1 NPRACH

Vysílání preamble s náhodným přístupem je prvním krokem procedury s přístupem, která umožňuje uživatelskému zařízení navázat spojení se sítí. Kromě přesné detekce preamble je odhadování časování UL dalším hlavním cílem přijímače NPRACH. Požadované parametry NPRACH konfigurované vyššími vrstvami jsou znázorněny na obrázku 2.3 a jsou definovány následovně [2].

- $N_{\text{period}}^{\text{NPRACH}}$ (3 bits, NPRACH-Periodicity)
- $N_{\text{rep}}^{\text{NPRACH}}$ (3 bits, NPRACH-NumRepetitions)
- $N_{\text{scoffset}}^{\text{NPRACH}}$ (3 bits, NPRACH-SubcarrierOffset)

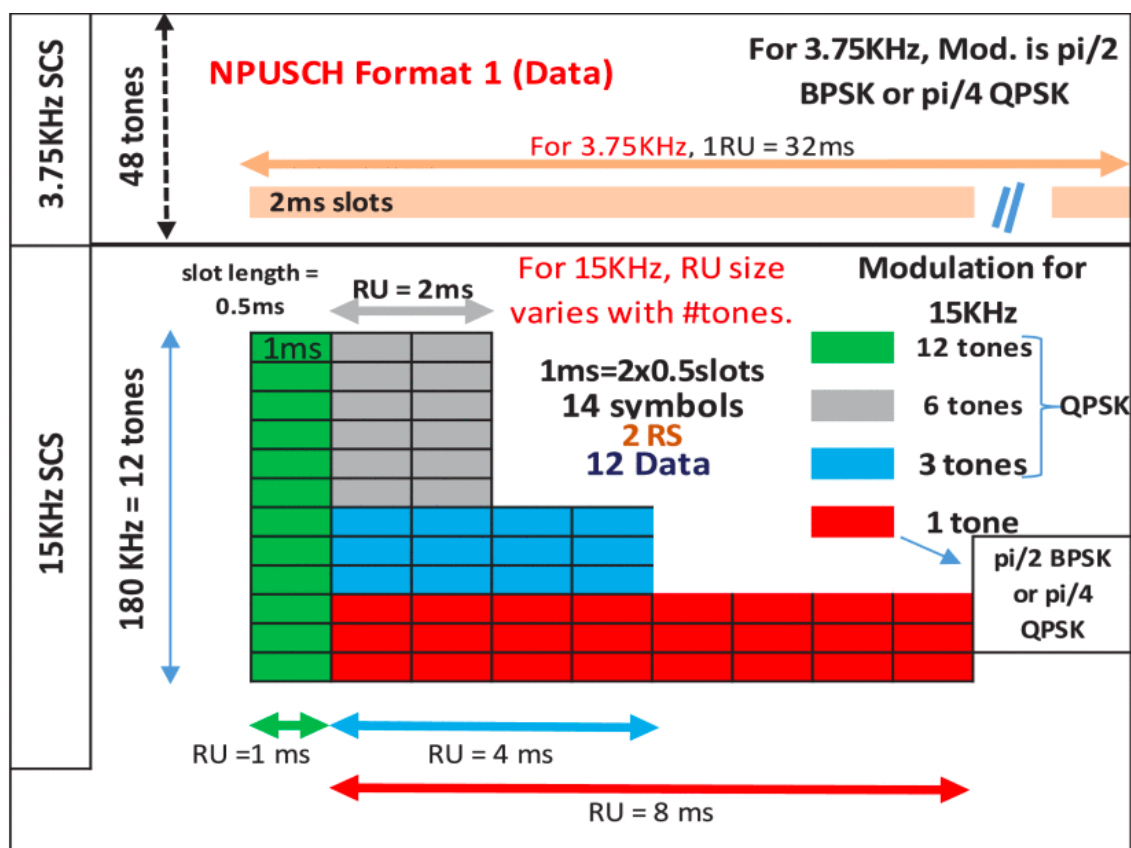
- N_{sc}^{NPRACH} (2 bits, NPRACH-NumSubcarriers)
- N_{start}^{NPRACH} (3 bits, NPRACH-StartTime)



Obr. 2.3: Struktura času a frekvence NPRACH, převzato z [2]

2.3.2 NPUSCH formát 1 (Data)

Data sdíleného kanálu NB-IoT UL (uplink) jsou mapována do formátu NPUSCH 1 a podporují 1, 3, 6 nebo 12 tónů. Vzhledem k tomu, že šířka pásma je omezena na 180 kHz, jsou definovány zdrojové jednotky přidělované v čase, tzv. PRB (physical resource block). Podrobnosti o délce jednotky zdroje v ms, délce slotu v ms, použité modulaci, počtu symbolů a referenčních signálech jsou uvedeny na obrázku 2.4. NPUSCH formát 1 má jeden pilotní symbol na slot, používá turbo kódování a používá buď $\frac{\pi}{4}$ QPSK nebo $\frac{\pi}{2}$ BPSK modulaci. NPUSCH Formát 1 se používá k přenosu dat UL (uplink), které se zpracovávají ve formě transportních bloků (TB) [2].



Obr. 2.4: Struktura zdrojových jednotek NPUSCH Format 1 (data), převzato z [2]

2.3.3 NPUSCH formát 2 (Řízení)

Kanál NPUSCH Format 2 nese potvrzení DL (downlink) dat a podporuje pouze jeden tón s roztečí subnosných 3,75 nebo 15 kHz. Kanál NPUSCH Format 2 má dobu trvání RU (resource unit) 2 ms a má tři referenční symboly na slot, což usnadňuje spolehlivé odhady. Jediný tón NPUSCH formát 2 je podobný tónu NPUSCH formátu 1, s výjimkou pilotní struktury, kódování kanálu a modulačního schématu. NPUSCH Format 2 má tři pilotní symboly na slot, používá opakovací kódování na jednom ACK / NACK bitu a používá pouze $\frac{\pi}{2}$ BPSK modulační schéma [2].

2.4 Technická specifikace

2.4.1 Vysílací výkon

Ve srovnání se staršími zařízeními GPRS je vysílací výkon rozšířen o 20 dB a to odpovídá dosažení MCL (Maximum Coupling Loss) 164 dB. Cílem je vylepšit energetickou účinnost jako například poskytnout životnost baterie okolo deseti let s kapacitou baterie 5 Wh při 164 dB MCL. Latence neboli zpoždění zprávy se pohybuje

okolo 10 sekund nebo méně [13].

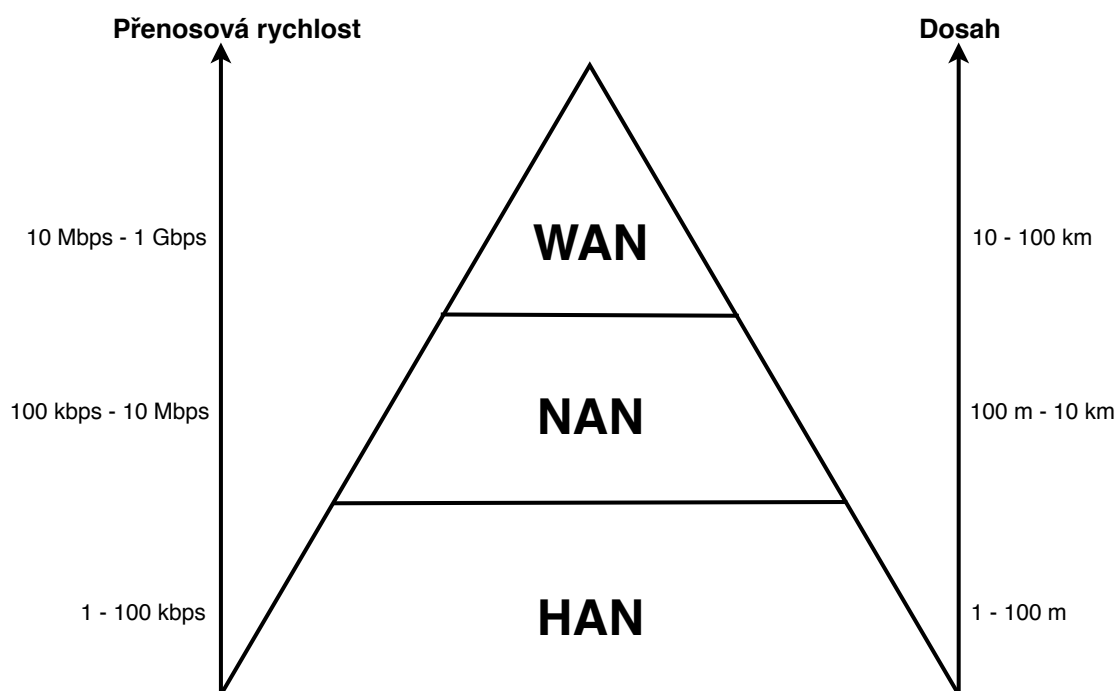
2.4.2 Frekvence

Ve standalone provozu může NB-IoT obsadit jeden GSM kanál s šířkou pásma 200 kHz, zatímco pro In-Band a Guard Band provoz je použit jeden blok fyzických zdrojů LTE s šířkou pásma 180 kHz [13].

V eMTC (enhanced Machine Type Communications) je zavedeno uživatelské zařízení se sníženou šířkou pásma rádiových frekvencí 1,4 MHz v downlink a uplink směru [13].

2.4.3 Teoretický dosah

Komunikační sítě můžeme klasifikovat podle rychlosti přenosu dat a rozsahu pokrytí, což představuje hierarchickou vícevrstvou architekturu, tj. HAN (Home Area Network), NAN (Neighborhood Area Networks) a WAN (Wide Area Network), tak jak jsou zobrazené na obrázku 2.5. NB-IoT může být optimálně použito například v NAN [11], ale i v ostatních oblastech se najde využití. Jeho maximální dosah je okolo 15 km, tak jak je uvedeno v tabulce 2.1 [20].



Obr. 2.5: Datová rychlost a komunikační rozsah pro komunikační sítě, převzato z [11], upraveno

2.4.4 Pokrytí

Podpora velkého počtu zařízení s nízkou propustností je cílem NB-IoT. Uvnitř buňky sítě je podpora alespoň okolo 52 000 zařízení. Každá země má svoje stanovené parametry pro množství zařízení na domácnost. Pro Londýn byla tato hodnota stanovena na 40 [13].

2.4.5 Datová rychlost

U již zmiňovaného MCL by měla být v aplikační vrstvě podporována rychlost přenosu dat alespoň 160 b/s pro uplink i downlink [13]. Špičkové datové rychlosti NB-IoT jsou schopné dosáhnout až 230 kb/s pro downlink a 250 kb/s pro uplink [11]. V praxi se využívá převážně datová rychlost, která je uvedena v tabulce 2.1.

2.4.6 Souhrn technických parametrů

Níže je uvedena tabulka 2.1 základních parametrů a informací ohledně technologie NB-IoT.

Tab. 2.1: Souhrn technických parametrů NB-IoT, převzato z [20]

Technologie	NB-IoT
Šířka pásma	180 kHz
Spektrum	In-Band GSM
Životnost baterie	10 let
Uplink	0,3–62,5 kbps
Downlink	0,5–27,5 kbps
Payload UL/DL	1600 B
Dvoucestná data	Half / Full duplex
Zabezpečení	3GPP (128-256)
Škálovatelnost	Vysoká
Dosah (km)	15
Technická podpora	Ne
Max. ERP	23 dBm
Nejlepší citlivost	−142 dB
Minimální náklady	10 \$
Společnost	3GPP

3 Technologie LTE Cat-M1

3.1 Úvod do technologie

LTE Cat-M1 (Long Term Evolution Category M1) je technologie, která je součástí standardu 3GPP (Verze 13) podporující LPWAN sítě v licencovaném pásmu. LTE Cat-M1 se zaměřuje na IoT aplikace s nízkou až střední datovou propustností, a také na zařízení vyžadující dlouhou životnost baterie [3].

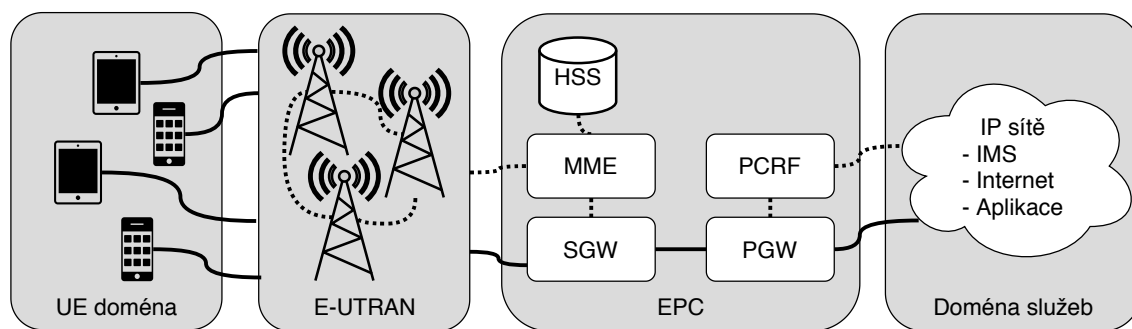
Další výhodou je podpora hlasových funkcí prostřednictvím VoLTE (Voice over LTE), což znamená, že ji lze použít pro aplikace vyžadující úroveň lidské interakce, například pro určité aplikace v oblasti zdraví a zabezpečení [20].

Skutečnou výhodou LTE Cat-M1 oproti jiným možnostem je obecně kompatibilita se stávající sítí LTE [24].

Tato technologie pokrývá různé aplikace v mnoha oblastech jako jsou například inteligentní domácnost, bezpečnostní systémy, průmyslové monitorování a řízení, sledování majetku, telematika, propojené zdraví, inteligentní měření, chytrá města a nositelná zařízení [3].

3.1.1 Architektury sítě LTE

Sítě LTE se skládají ze dvou hlavních částí, a to E-UTRAN (Evolved - UMTS Terrestrial Radio Access Network) a EPC (Evolved Packet Core). Kromě toho mají LTE sítě UE (User Equipment – Uživatelské Zařízení) doménu a doménu služeb jako jsou například sítě IP. Příklad můžeme vidět na obrázku 3.1 [21].



Obr. 3.1: Síťová architektura LTE, převzato z [21], upraveno

E-UTRAN, část rádiové přístupové sítě, se skládá ze základních stanic, které jsou známé jako eNBs (evolved NodeBs). Zařízení eNB je zodpovědné za spolehlivý vysokofrekvenční přenos mezi základní sítí LTE a UEs. V datové rovině jsou eNB zodpovědné za spolehlivé doručování napříč vysokofrekvenčním přenosem, šifrováním

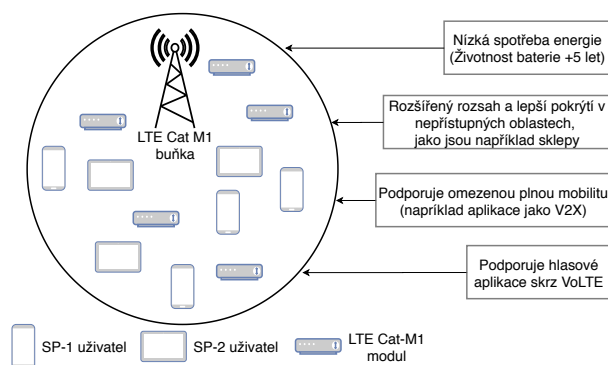
dat rádiového rozhraní, kompresi hlaviček a ochranu integrity prostřednictvím zásobníku rádiových protokolů. V řídicí rovině jsou eNB odpovědné za správu rádiových zdrojů, včetně řízení přístupu, řízení rádiových nosičů a plánování prostřednictvím zásobníku protokolu rádiových zdrojů [21].

EPC, hlavní síťová část, se skládá z SGW (Serving GateWay), PGW (PDN (Packet Data Network) GateWay), MME (Mobile Management Entity), HSS (Home Subscriber Server) a softwarový uzel PCRF (Policy and Charging Rules Function) [21].

- SGW, hlavní část EPC, je základním prostředkem mezi E-UTRAN a EPC. Navíc pomáhá při spolupráci s dalšími technologiemi 3GPP, jako jsou systémy GPRS (General Packet Radio Service) nebo UMTS (Universal Mobile Telecommunication System) [21].
- MME je řídicí uzel, který zpracovává signalizaci mezi UE a EPC. MME se rovněž stará o relaci a správu mobility v řídicí rovině [21].
- PGW je zodpovědný za poskytování IP připojení k UEs, přidělování IP k jednotlivým UEs, připojení LTE sítě k jiným PDNs a zajišťování kvality služeb (QoS – Quality of Service) prostřednictvím PCRF [21].
- HSS je zodpovědný za uchovávání informací uživatelů, včetně předplatného, uchovávání informací o PDNs, přenášení ověřovacích funkcí a pomoc při řízení mobility [21].

3.2 Systém model

Předpokládejme bezdrátovou síť se systémem LTE, který se skládá z jedné základnové stanice LTE, sadu poskytovatelů služeb (SP – Service Provider), z nichž každý poskytuje a zajišťuje služby uživatelům. Kromě toho má síť N zařízení IoT, z nichž každé je vybaveno modemem LTE Cat-M1. Popsaný model je vyobrazen v obrázku 3.2 [24].



Obr. 3.2: Systém model LTE Cat-M1, převzato z [24], upraveno

Aby bylo možné dosáhnout koexistence mezi LTE a Cat-M, předpokládá se minimální rušení. Proto při přiřazování zdrojového bloku (RB – Resource Block) r^{th} k poskytovateli služeb LTE s^{th} uživateli u je dána propustnost downlinku

$$T(u, r, s) = B \cdot \log_2 \left(1 + \frac{P_{r,u}^s G_{BS,u}^{s,r}}{\sigma^2} \right),$$

kde B je šířka pásma RB v jednotkách Hz, $P_{r,u}^s$ je přenosový výkon DL mezi LTE-BS a LTE poskytovatele služeb s^{th} k uživateli u , $G_{BS,u}^{s,r}$ je zisk kanálu mezi LTE-BS a LTE poskytovatele služeb s^{th} k uživateli u na zdrojovém bloku (RB) r^{th} a σ^2 je rozptyl šumu [24].

Celková propustnost LTE je tedy dána vztahem

$$T_{lte} = \sum_{s=1}^S \sum_{u=1}^U \sum_{r=1}^R a(s, u, r) \cdot T(u, r, s),$$

kde binární koeficient $a(s, u, r) = 1$, když je zdrojový blok r^{th} přiřazen LTE poskytovateli služeb s^{th} k uživateli u a jinak je $a(u, r, s) = 0$ [24].

Podobně je propustnost zařízení IoT m při připojení k zdrojovému bloku (RB) r^{th} je dána vztahem

$$T(r, m) = B \cdot \log_2 \left(1 + \frac{P_{r,m} G_{r,m}}{\sigma^2} \right),$$

kde $P_{r,m}$ je přenosový výkon DL mezi LTE-BS a zařízením IoT m^{th} , $G_{r,m}$ je kanálový zisk mezi LTE-BS a zařízením IoT m na zdrojovém bloku (RB) r [24].

Podle toho je celková propustnost IoT Cat-M dána vztahem

$$T_{IoT} = \sum_{m=1}^M \sum_{r=1}^R a(r, m) \cdot T(r, m),$$

kde binární koeficient $a(r, m) = 1$, když je zařízení IoT m připojeno ke zdrojovému bloku (RB) r^{th} a jinak je $a(r, m) = 0$ [24].

Nakonec pro výpočet celkové propustnosti sítě se přidá propustnost LTE k propustnosti IoT. Podle toho je celková propustnost sítě dána vztahem

$$T = T_{lte} + T_{IoT}$$

[24].

Jak již bylo zmíněno dříve, pro koexistenci mezi uživateli LTE a zařízeními IoT se předpokládá minimální až ideálně žádné rušení. Dále se předpokládá, že všichni uživatelé sítě LTE a zařízení IoT mají vždy data k odeslání. Hlavním cílem plánovače koexistence je najít nejlepší algoritmus přiřazení nebo plánování RBs, který maximalizuje celkovou propustnost sítě. Za tímto účelem existují tři různé scénáře koexistence, které závisí na různých prioritách LTE a IoT [24].

3.2.1 Scénář 1: Vyšší priorita provozu IoT

V tomto scénáři se předpokládá, že provoz IoT má vyšší prioritu než provoz LTE. Jinými slovy, jakmile bude do sítě zavedeno více zařízení IoT, bude provozu IoT přiděleno více zdrojových bloků (RB – Resource Block) na úkor provozu LTE [24].

3.2.2 Scénář 2: Vyšší priorita LTE provozu

Tento scénář je opakem scénáře číslo 1, ve kterém má přenos LTE vyšší prioritu než přenos IoT. Jinak řečeno, jakmile bude do sítě zavedeno více zařízení LTE, bude provozu LTE přiřazeno více zdrojových bloků (RB) na úkor provozu IoT [24].

3.2.3 Scénář 3: Provoz IoT i provoz LTE mají stejnou prioritu

Tento scénář je zobecněným případem, ve kterém je plánování zdrojových bloků (RB) zaměřeno na maximalizaci celkové propustnosti sítě při stejné prioritě pro oba provozu LTE a IoT [24].

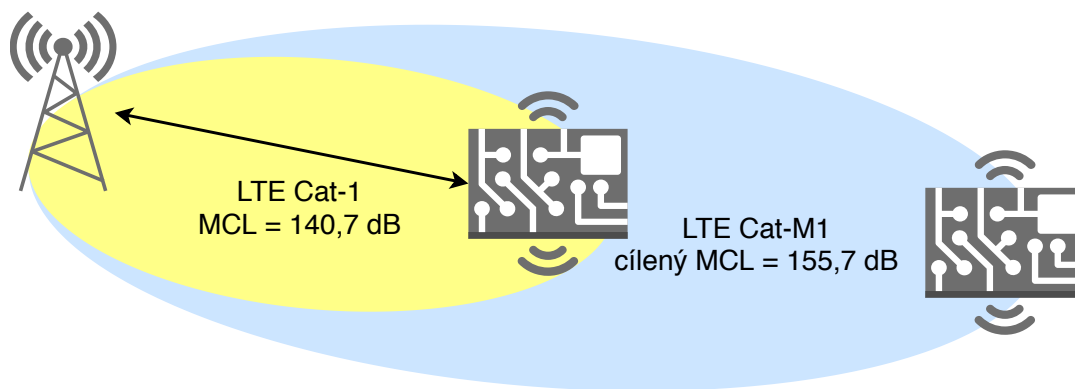
3.3 Pokrytí

Jako jedna z pokročilých síťových technologií je LTE vhodné pro připojení velkého množství zařízení MTC (Machine Type Communication) do sítě. Avšak současná infrastruktura sítí LTE je postavena hlavně pro širokopásmovou komunikaci používanou chytrými telefony [7].

LTE Release 13 konkrétně definuje novou kategorii UE, konkrétně kategorii M1 (známou jako LTE Cat-M1), aby bylo dosaženo vylepšeného pokrytí při snížené šířce pásma 1,4 MHz. Aby se dále snížily náklady a složitost zařízení, tak LTE Cat-M1 podporuje provoz pouze s jednou přijímací anténou. S nízkou složitostí hardwarového designu se zvyšuje náročnost na podmínky a dostatečné pokrytí sítě [7].

Pokrytí se obvykle měří pomocí metriky MCL (Maximum Coupling Loss), která slouží jako jednotná míra pro různá frekvenční pásma a prostředí. Což je parametr, který je definován minimální přijatelnou přijímanou úrovní výkonu, respektive udává maximální dovolenou ztrátu výkonu signálu, kterou dokáže systém tolerovat a zůstat při ní provozuschopný. Ve srovnání se starší technologií LTE Cat-1 si verze 13 u LTE Cat-M1 klade za cíl zvýšit pokrytí nejméně o 15 dB, tj. ze 140,7 dB na 155,7 dB [7]. Srovnání pokrytí v důsledky změny MCL lze vidět na obrázku 3.3.

Ve verzi 13 se zlepšení pokrytí dosahuje hlavně pomocí opakování, tj. stejný transportní blok se vysílá ve více po sobě jdoucích dílčích rámcích, aby se zvýšila přijímaná energie na informační bit. Během náhodného přístupu je počet opakování konfigurován na základě přijímaného výkonu referenčního signálu (RSRP – Reference Signal



Obr. 3.3: Vylepšení cílového pokrytí LTE Cat-M1, převzato z [7], upraveno

Received Power) měřeného UE. Teoreticky se očekává, že zisk pokrytí poroste lineárně s počtem opakování, když lze pro souvislou demodulaci perfektně odhadnout odezvu kanálu. Odezvu kanálu však v některých případech není možné dokonale určit. Proto efektivní zisk pokrytí poskytovaný schématem opakování silně závisí na skutečné implementaci odhadu kanálu a měření [7].

3.3.1 Provozní režimy pro vylepšené pokrytí

1. CE režim

Verze 13 podporuje dva provozní režimy pro vylepšení pokrytí, konkrétně je to režim CE A a režim CE B. Režim A je hlavně pro zařízení s mírným požadavkem na pokrytí, zatímco režim B je určen pro větší oblast pokrytí. Provozní režim je konfigurován eNB (evolved Node B) během RRC (Radio Resource Control) připojení. V každém provozním režimu A a B (Verze 13) jsou dále definovány dvě úrovně CE, aby poskytovaly ještě jemnější úrovně pokrytí. Rozdíl od operačního režimu je v tom, že CE úroveň je vybrána podle UEs založených na přijaté síle referenčních signálu tzv. RSRP (Received Power of Reference Signals) a poté oznámena k eNB během náhodného přístupu. Tím se účinně předejde zbytečnému rušení a usnadní se postup náhodného přístupu bez plýtvání síťovými prostředky [7].

2. MPDCCH a plánování mezi dílčími rámci

Kvůli snížené šířce pásma LTE Cat-M1 nelze konvenční řídicí kanál PDCCH (Physical Downlink Control Channel), který přenáší řídicí zprávy v celém pásmu, dekodovat úzkopásmovými zařízeními LTE Cat-M1. Verze 13 proto zavádí speciální řídicí kanál s názvem MPDCCH, který zabírá celé dílčí rámce pro doručování řídicích zpráv. LTE Cat-M1 tedy využívá plánování napříč díl-

čími rámci, tj. plánovací informace přenášené MPDCCH a užitečné zatížení nesené PDSCH (Physical Downlink Shared Channel) nejsou ve stejném dílčím rámci. Navíc pro dosažení rozšířeného pokrytí je ve verzi 13 pro MPDCCH k dispozici také schéma opakování [7].

3. Počet opakování

Verze 13 definuje sadu dostupných opakovacích čísel pro různé fyzické kanály v různých režimech CE. Pro PDSCH i PUSCH (Physical Uplink Shared Channel) v CE režimu A jsou dostupná čísla opakování 1, 2, 4, 8, 16 a 32. Na druhou stranu pro CE režim B lze maximální opakovací číslo nakonfigurovat až do hodnoty 2048 pro velmi hluboké pokrytí. Pro MPDCCH opakovací číslo se pohybuje od 1 do 256 v obou CE režimech A i B [7].

3.4 Technická specifikace

3.4.1 Vysílací výkon

LTE Cat-M1 dosahuje MCL hodnoty okolo 155,7 dB [7]. Výdrž baterie se pohybuje okolo 10 let na jedno nabití. V některých případech při větší spotřebě energie v důsledku většího vytížení zařízení může docházet ke snížení výdrže baterie. Instalace baterie v některých případech přispívá ke snížení nákladů na údržbu nasazených zařízení, a to i v místech, kde nemusí být koncová zařízení připojena přímo k elektrické síti [20].

3.4.2 Frekvence

Maximální šířka pásma systému pro LTE Cat-M1 je 1,4 MHz oproti 20 MHz u LTE Cat-0 a Cat-1 [24].

3.4.3 Teoretický dosah

Teoretický dosah podle tabulky 3.1 se pohybuje okolo 11 km. Když vzdálenost mezi zařízeními M2M (Machine to Machine) a eNodeB dosáhne 12 km, propustnost se u technologie LTE Cat-M1 začne snižovat. U LTE-M je rychlost poklesu propustnosti daleko menší oproti starším technologiím, a proto dosahuje širší vzdálenosti. Pokud se budeme bavit o maximální vzdálenosti, kde zařízení může stále přenášet data, bavíme se o vzdálenosti 60 km [4].

K roku 2020 byl odhad připojených zařízení cca 25 miliard [4].

3.4.4 Datová rychlost

S rychlostí uplink a downlink 375 kb/s v režimu polovičního duplexu LTE Cat-M1 konkrétně podporuje aplikace IoT se spotřebou nízké až střední rychlosti přenosu dat. V tabulce 3.1 je uvedena datová rychlost 1 Mb/s, kterou tato technologie také disponuje. Při těchto rychlostech může LTE Cat-M1 poskytovat vzdálené aktualizace firmwaru bezdrátově (FOTA – Firmware Over The Air) v rozumných časových rámcích, takže je vhodný pro kritické aplikace běžící na zařízeních, která mohou být nasazena v terénu po delší dobu [20].

3.4.5 Souhrn technických parametrů

Níže je uvedena tabulka 3.1 základních parametrů a informací ohledně technologie LTE Cat-M1.

Tab. 3.1: Souhrn technických parametrů, převzato z [20]

Technologie	LTE Cat-M1
Šířka pásma	1,08 MHz
Spektrum	In-Band GSM
Životnost baterie	5 let
Přenosová rychlost	1 Mbps
Payload UL/DL	1600 B
Dvoucestná data	Half duplex
Zabezpečení	3GPP (128-256)
Škálovatelnost	Vysoká
Dosah (km)	11
Nejlepší citlivost	−123 dB
Minimální náklady	20 \$
TX spotřeba	100–490mA
Společnost	3GPP

4 Srovnání NB-IoT a LTE Cat-M1

Tato kapitola pojednává o komplexním srovnání dvou popisovaných technologií z různých hledisek. Shrnutí je kromě porovnání výběr jedné z technologií, která bude v další kapitole podrobena simulaci v uvažovaných aplikacích SmartGrids.

4.1 Spotřeba

Vzhledem k tomu, že zařízení NB-IoT se díky své konektivitě spoléhají na jednoduché tvary vln, spotřebovávají mnohem méně energie než LTE Cat-M1. To však také znamená, že zařízení NB-IoT nemohou odesílat tolik dat jako LTE Cat-M1 a nejsou schopna spolupracovat s jinými mobilními zařízeními [12].

Zatímco u NB-IoT je všeobecně akceptovanou metou životnost baterie 10 let, u LTE-M se reálně hovoří o jednotkách let nebo o podpoře ze zdroje zařízení, případně o solárním napájení [19].

4.2 Dostupnost služeb v globálním kontextu

Například v USA se nasazení NB-IoT potýká s problémy, protože převahu tam drží LTE infrastruktura. Zato například v Asii, zejména v Číně, je situace příznivější [12].

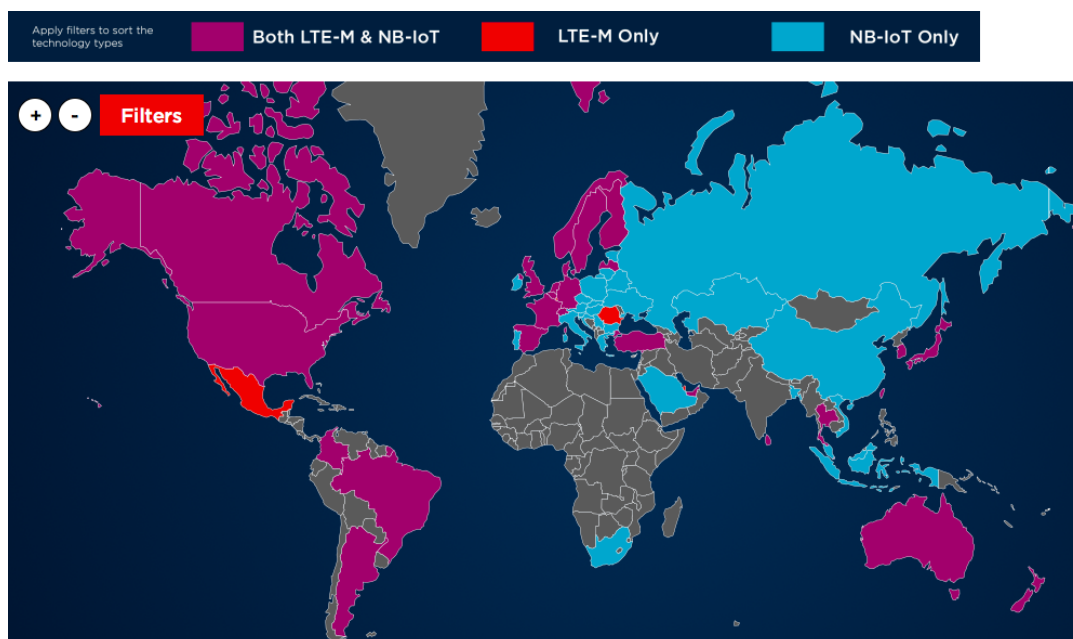
V Asii se ujala varianta NB-IoT, využívající jako nosič síť WCDMA. V USA je NB-IoT k dispozici pouze u operátora T-Mobile [19].

Naopak v některých částech Asie, Afriky, ale i Evropy mohou nastat problémy s pokrytím LTE Cat-M1 zařízení z důvodu starší generace infrastruktury postavené převážně na 3G [12].

Evropě dominuje technologie NB-IoT, výjimku tvoří jen několik případů:

- Francie, kde je k dispozici zatím pouze LTE-M,
 - Belgie s NB-IoT i LTE-M (Orange)
 - a Nizozemí s NB-IoT i LTE-M (díky konkurenci T-Mobile, Vodafone a KPN)
- [19].

Níže je obrázek 4.1, který popisuje zastoupení technologií LTE-M a NB-IoT ve světě v roce 2020. V některých zemích jsou podporované pouze LTE-M nebo NB-IoT, a naopak některé podporují obě technologie. V tabulce 4.1 jsou vypsáni někteří operátoři dle geografické lokality a použité technologií. Webová stránka GSMA uvádí podrobnější informace o ostatních operátorech, např. Orange nebo Telefonica [5].



Obr. 4.1: Mapa zastoupení technologie NB-IoT a LTE Cat-M1 ve světě 2020, převzato z [6]

Tab. 4.1: Výpis operátorů 2020 (umístění a použité technologie), převzato z [5]

Počet LTE-M sítí		Počet NB-IoT sítí	Celkový počet IoT sítí
48		101	149
Operátor	Země/Region		Technologie
AT&T	Mexico		LTE-M
	Spojené státy americké		
	Spojené státy americké		NB-IoT
Deutsche Telekom	Česká republika (T-Mobile CZ)		NB-IoT
	Polsko (DT Polska)		
	Slovensko (Slovak Telecom)		
	Spojené státy americké (T-Mobile US)		
	Německo (Telekom Deutschland)		LTE-M & NB-IoT
Vodafone	Česká republika		NB-IoT
	Německo		
	Itálie		
	Velká Británie		
	Nový Zéland		
	Nizozemsko		LTE-M & NB-IoT

4.3 Rozsah pokrytí

Konektivita LTE Cat-M1 i NB-IoT zvládne široký dosah, avšak pro projekty IoT s extrémně velkou oblastí pokrytí je NB-IoT vhodnou volbou konektivity a přináší sedmkrát větší rozsah současných technologií LTE, jako je Cat-M1 [12].

Zařízení NB-IoT vynikají ve vnitřním průmyslovém prostředí a poskytují podnikům pokrytí více než 20 dB vysílacího výkonu. Protože se zařízení NB-IoT spoléhají na jednoduchou formu signálu a přenášejí jen velmi malé množství dat, mohou nabídnout lepší cestu signálu a pronikání přes překážky. To poskytuje podnikům větší rozsah a pokrytí ideální pro projekty IoT, které mají senzory rozmístěné v podzemí nebo v jiných těžko dostupných oblastech se špatným signálem [12].

4.4 Připojení

Rozhodování o tom, které IoT připojení by bylo pro individuální projekt nejlepší je velmi důležité. Například pokud sledujeme síťový provoz v reálném čase, když paket cestuje z jednoho místa do druhého, může být optimální volbou připojení LTE Cat-M1. Protože LTE Cat-M1 může plynule přepínat z jedné základnové stanice do druhé, aniž by došlo k přerušení připojení. Na druhé straně, pokud je datový přenos přerušen i na menší okamžik ztraceným signálem nebo pohybem zařízení z místa, přenos bude zastaven, dokud nebude zařízení znovu připojeno [12].

4.5 Náklady

Náklady jsou nedílnou součástí každého projektu, a zatímco u starších sítí 2G a 3G s GPRS a dalšími standardy pro přenos dat byly podmínky dlouhodobější a stálejší pro všechny uživatele stejné, tak u NB-IoT a LTE-M lze předpokládat rozdíly. Tam, kde rozhoduje množství přenesených dat, je vhodné zaměřit se na tento faktor. V mnoha případech je konvenční přenos dat jednoho Mb až 100x levnější než využití LPWAN [19].

4.6 Shrnutí

Technicky je volba poměrně jednoduchá – přenosová rychlost a latence určuje vhodnost pro jednotlivé aplikace. K parametrům přenosu lze dodat ještě fakt, že NB-IoT není vhodné pro „úplně mobilní aplikace“, tedy pro ty, kde se předpokládá přenos dat při pohybu zařízení. Obě aplikace lze sice označit jako mobilní, ale rozdíl je značný [19].

V konečném důsledku bylo k simulaci vybráno LTE Cat-M1, kterému se věnuje zbytek práce, protože hledám technologii se schopnosti

- přenosu většího objemu dat,
- krátkého dosahu signálu,
- možnost měření provozu v reálném čase.

5 Network Simulator 3

5.1 Úvod do NS-3

NS-3 (Network Simulator 3) je bezplatné softwarové simulační prostředí zaměřené na síťové technologie [26]. Je podporováno hlavně pro systémy Linux, OS X a FreeBSD [21]. Využívá se pro vývoj a návrh síťových technologií. NS-3 obsahuje množství modulů souvisejících prakticky s většinou všemi prvky síťových technologií. NS-3 projekt je open-source projekt, který není rozšířením staršího simulátoru NS-2, jedná se o nový simulátor [26].

Oba simulátory jsou napsány v C++, ale NS-3 nepodporuje API NS-2. Některé modely z NS-2 již byly přeneseny do NS-3. Změna oproti NS-2 je ve volbě skriptovacího jazyka, a proto je NS-3 napsáno výhradně v jazyce C++ s volitelnými vazbami na Python [26]. Kódy pro simulaci různých scénářů lze vyvíjet pomocí programovacích jazyků C++ nebo Python [21].

Výsledky samotných simulací lze zobrazit pomocí NetAnim, a to vygenerováním souboru ve formátu xml. Vzhledem k tomu, že NS-3 generuje trasovací soubory paketů pcap, lze k analýze tras použít také další nástroje například program Wireshark.

Všechny NS-3 mají následující funkce jako jsou [26]:

- C++ programy a skriptování v jazyce Python,
- sladění se skutečnými systémy (zásuvky, rozhraní ovladačů zařízení),
- důraz na integraci softwaru,
- virtualizace a integrace testovacího zařízení jsou prioritou (režimy emulace),
- dobře zdokumentovaný systém atributů
- a aktualizované modely.

5.2 Modul LENA

Modul LENA byl vytvořen jako produktově orientovaná simulace, která výrobcům zařízení LTE dává možnost testovat algoritmy správy v realistických scénářích. Modul LENA je open-source simulátor, který umožňuje návrh, vyhodnocení a ověření technologie LTE. LENA byla oficiálně vydána v roce 2011. Modul LENA se skládá ze dvou částí: model LTE a model EPC [21].

Model LTE zahrnuje sadu rádiových protokolů, které spočívají ve vzdušném rozhraní bezdrátové přístupové sítě mezi UE (User Equipment) a eNB (evolved NodeBs). Zásobník rádiových protokolů se skládá z vrstev fyzického, vícenásobného přístupu (MAC – Multiple Access Control), řízení rádiového spojení (RLC – Radio Link Control), paketového datového konvergenčního protokolu (PDCP – Packet

Data Convergence Protocol) a řízení rádiových zdrojů (RRC – Radio Resource Control) [21].

Zatímco model EPC je všemi rozhraními umístěn na straně hlavní sítě tzv. Core Network Side. Tato rozhraní existují mezi eNB, PGW, S-GW, MME a dalšími entitami hlavní sítě. Konektivita IP typu end-to-end je poskytována v rámci modelu EPC prostřednictvím P-GW [21].

5.3 Možné výsledky z simulací

Výsledky z NS-3 můžeme zhodnotit v několika bodech jako například

- specifikace kanálu,
- posouzení síly signálu,
- testování plánovače MAC,
- test mobility a nebo
- režim přenosu [21].

6 Scénáře z NS-3

V této kapitole jsou popsány dva scénáře pro technologii LTE a jeho tvorba v prostředí NS-3 s využitím modulu LENA. Kapitola se dělí na dvě sekce, které popisují simulované sítě, a to pro počáteční (6.1) a rozšířený (6.2) scénář. V jednotlivých sekcích je popsána funkčnost a topologie scénáře.

6.1 Počáteční scénář NS-3

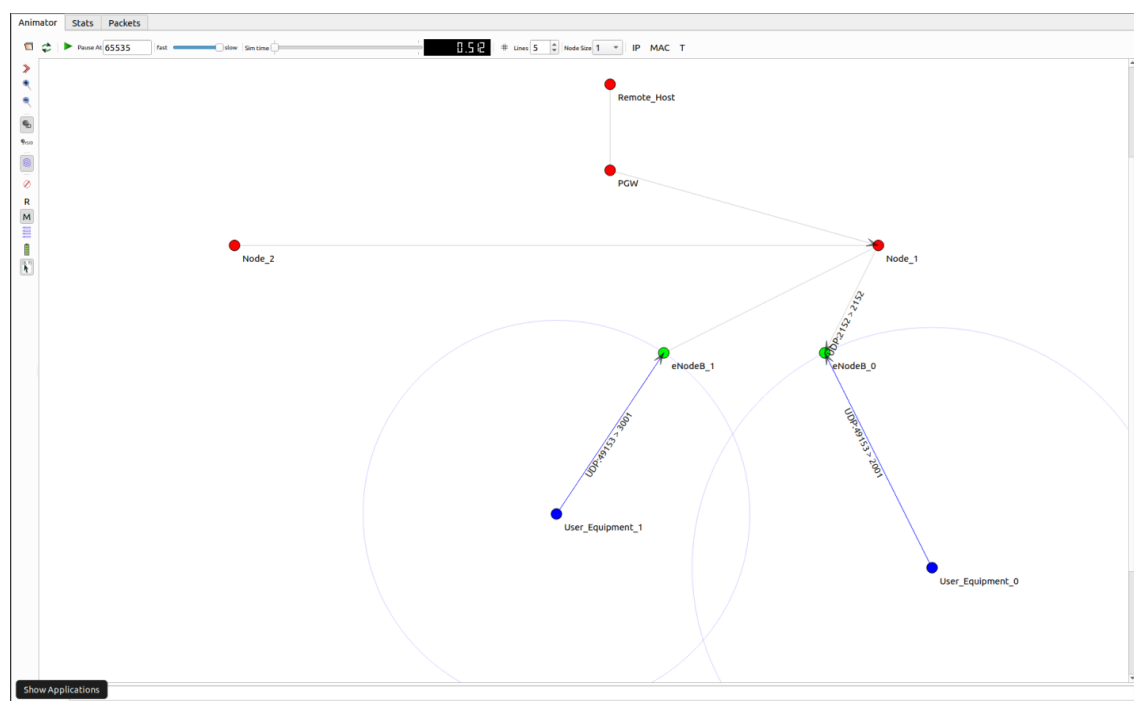
Výchozím zdrojem informací byla ukázka scénáře „lena-simple-epc.cc“, která je veřejně dostupná v dokumentaci NS-3 [16]. Scénář má ukázat funkčnost vytvořené sítě, možnost stavět nové aplikace na této síťové architektuře a rozvíjet topologii o nová zařízení.

Tento základní scénář byl dále upraven v těchto bodech:

- úprava některých částí kódu,
- nastavení topologie sítě,
- přidání nových knihoven a funkcí
- a implementace metod pro měření sítě.

Ve scénáři byla použita původní aplikace ze scénáře „lena-simple-epc.cc“, která běží v samotné síti. Byla zde použita komunikace založená na UDP paketů.

Topologie scénáře je zobrazena na obrázku 6.1.



Obr. 6.1: Topologie sítě počátečního scénáře v programu NetAnim

Samotná topologie je tvořená několika zařízeními jako je

- Remote_Host (přistupuje k Internetu),
- PGW (je součástí EPC),
- Node_1 = SGW (je součástí EPC),
- eNodeB_0 a eNodeB_1 (jsou součástí E-UTRAN),
- User_Equipment_0 a User_Equipment_1.

6.1.1 Výpis kódu

Výpis 6.1: Definice proměnných na začátku

```
1  uint16_t numNodePairs = 2;
2  Time simTime = MilliSeconds (1100);
3  double distance = 60.0;
4  Time interPacketInterval = MilliSeconds (100);
5
6  bool useCa = false;
7  bool disableDl = false;
8  bool disableUl = false;
9  bool disablePl = false;
10 bool enablePcap = true;
11 bool enableAscii = false;
```

V kódu 6.1 jsou uvedené proměnné, které zasahují do funkčnosti programu. Proměnná `numNodePairs` definuje počet pro zařízení UE a eNodeB. V tomto případě se vytvoří dvě eNodeB a ke každému bude přiřazeno jedno UE.

Proměnná `simTime` nastavuje dobu trvání simulace. Po tuto dobu dochází ke komunikaci mezi Remote_Host a UEs. Hodnoty typu „bool“ mají funkci zapínání nebo vypínání funkcí v programu.

Výpis 6.2: Vytvoření uzlů v síti

```
1  Ptr<Node> pgw = epcHelper->GetPgwNode ();
2
3  NodeContainer remoteHostContainer;
4  remoteHostContainer.Create (1);
5  Ptr<Node> remoteHost = remoteHostContainer.Get (0);
6  InternetStackHelper internet;
7  internet.Install (remoteHostContainer);
8
9  NodeContainer ueNodes;
10 NodeContainer enbNodes;
11 enbNodes.Create (numNodePairs);
```

```
12 ueNodes.Create (numNodePairs);
```

Ve výpisu 6.2 jsou uvedené příkazy, které vytváří jednotlivé uzly (PGW, Remote_Host, UEs a eNodeB). K vytváření uzlů se používá třída `NodeContainer`, která v sobě obsahuje i samotný počet vytvoření jednotlivých uzlů. To se provádí příkazem „<název-uzlu>.Create (<počet-uzlů>)“. U vytvoření uzlu PGW se využívá knihovna „epc-helper.h“ a zde se volá metoda `GetPgwNode()`.

Výpis 6.3: Vytvoření internetu

```
1 PointToPointHelper p2ph;  
2 p2ph.SetDeviceAttribute ("DataRate",  
3 p2ph.SetDeviceAttribute ("Mtu",  
4 p2ph.SetChannelAttribute ("Delay",  
5 NetDeviceContainer internetDevices = p2ph.Install (pgw,  
6 Ipv4AddressHelper ipv4h;  
7 ipv4h.SetBase ("1.0.0.0", "255.0.0.0");  
8 Ipv4InterfaceContainer internetIpIfaces = ipv4h.Assign  
9  
10 Ipv4Address remoteHostAddr = internetIpIfaces.GetAddress (1);  
11  
12 Ipv4StaticRoutingHelper ipv4RoutingHelper;  
13 Ptr<Ipv4StaticRouting> remoteHostStaticRouting =  
14 remoteHostStaticRouting->AddNetworkRouteTo (Ipv4Address
```

Za pomoci příkazu `p2ph.SetDeviceAttribute (...)` nastavujeme parametr internetu jako je například rychlost přenosu dat (`DataRate`) nebo maximální velikost IP datagramu (tzv. MTU – Maximum Transmission Unit). Pro nastavení jednotlivých kanálů mezi uzly je příkaz `p2ph.SetChannelAttribute (...)`, který dle výpisu 6.3 popisuje zpoždění při přenosu. Prostředí internetu je nainstalováno do uzlů PGW a Remote_Host, které jsou uvnitř kontejneru `internetDevices`. Pomocí tří tříd `Ipv4AddressHelper`, `Ipv4Address` a `Ipv4StaticRoutingHelper` je nastavené IP směrování v síti, které je statické.

Výpis 6.4: Instalace mobilního modulu

```
1 Ptr<ListPositionAllocator> positionAlloc = CreateObject<  
2 for (uint16_t i = 0; i < numNodePairs; i++)  
3 {  
4     positionAlloc->Add (Vector (distance * i, 20, 0));  
5 }  
6 MobilityHelper mobility;  
7 mobility.SetMobilityModel ("ns3::ConstantPositionMobilityModel"  
8 mobility.SetPositionAllocator(positionAlloc);
```



```
9 mobility.InstallAll ();
```

Z výpisu 6.4 lze poznat, že se jedná o manuální umístění uzlů tzv. konstantní umístění, které je nastaveno příkazem `mobility.SetMobilityModel(...)`. Mobilita je vztažena na všechny uzly pomocí příkazu `mobility.InstallAll()`.

Výpis 6.5: Instalace LTE zařízení na uzly

```
1 NetDeviceContainer enbLteDevs = lteHelper->InstallEnbDevice
2 NetDeviceContainer ueLteDevs = lteHelper->InstallUeDevice
```

Ve výpisu 6.5 jsou vloženy a zároveň vytvořeny LTE zařízení na jednotlivých uzlech (`enbNodes` a `ueNodes` jsou vytvořeny ve výpisu 6.2), které jsou v kontejneru `enbLteDevs` a `ueLteDevs`.

Výpis 6.6: Přiřazení jednoho UE k jednomu eNodeB

```
1 for (uint16_t i = 0; i < numNodePairs; i++)
2 {
3     lteHelper->Attach (ueLteDevs.Get(i), enbLteDevs.Get(i));
4 }
```

V sekci 6.6 je část kódu, který popisuje přidělení jednoho UE k jednomu eNodeB. Je tvořen cyklem `for`, který se opakuje v závislosti na počtu UEs, které jsou definované v parametru `numNodePairs`.

Výpis 6.7: Instalace a start aplikací

```
1 uint16_t dlPort = 1100;
2 ApplicationContainer clientApps;
3 ApplicationContainer serverApps;
4
5 for (uint32_t u = 0; u < ueNodes.GetN (); ++u){
6     if (!disableDl)
7     {
8         PacketSinkHelper dlPacketSinkHelper ("ns3::UdpSocketFactory"
9         serverApps.Add (dlPacketSinkHelper.Install (
10
11         UdpClientHelper dlClient (ueIpIface.GetAddress (u), dlPort);
12         dlClient.SetAttribute ("Interval", TimeValue (
13         dlClient.SetAttribute ("MaxPackets", UintegerValue (
14         clientApps.Add (dlClient.Install (remoteHost));
15     }
```

Ve výpisu 6.7 je popsána aplikace pouze pro přenos downlink. Dále v kódu je definován i provoz pro uplink a mezi rovnocennými UEs. Na začátku je definován

směrový port `dlPort` a kontejner `ApplicationContainer` s klientskou a serverovou aplikací. Celkově aplikace využívá UDP pakety, které se směřují na port s hodnotou `dlPort`. Zároveň na tomto přenosovém kanálu jsou nastaveny vlastnosti downlinku například jako maximální počet paketů a interval jednotlivého paketu.

Výpis 6.8: PCAP trasování

```
1  if (enablePcap)
2  {
3      p2ph.EnablePcapAll("lena-simple-epc");
4  }
```

Výpis 6.8 slouží ke generování souboru, který popisuje síťovou komunikaci v síti. Vytvořené soubory mají příponu `*.pcap` a dají se spustit například v programu Wireshark.

Výpis 6.9: Animace v NetAnim

```
1  AnimationInterface anim ("lena-simple-epc.xml");
2  anim.EnablePacketMetadata(true);
3
4  anim.SetConstantPosition (ueNodes.Get(0), 65, 50);
5  anim.UpdateNodeDescription (ueNodes.Get(0),
6  anim.UpdateNodeColor (ueNodes.Get(0), 0, 0, 255);
```

Ve výpisu 6.9 je použita třída `AnimationInterface`, která slouží pro generování souboru s příponou `*.xml` pro aplikaci NetAnim. NetAnim je využíván pro animace vytvořených sítí. Příkaz `anim.SetConstantPosition(...)` je pro nastavení konstantní pozice pro jednotlivé uzly, `anim.UpdateNodeDescription(...)` je pro nastavení názvu uzlu a `anim.UpdateNodeColor(...)` nastavuje barvu uzlu v RGB formátu. Tímto způsobem se nastavily všechny uzly jako jsou UEs a nebo eNodeB.

Výpis 6.10: Implementace FlowMonitor

```
1  Ptr<FlowMonitor> flowMonitor;
2  FlowMonitorHelper flowHelper;
3  flowMonitor = flowHelper.InstallAll();
4
5  flowMonitor->SerializeToXmlFile("FlowMonitor.xml",
```

Samotné řádky ve výpisu 6.10 vytváří metodu, která monitoruje síť a výsledky ukládá do souboru `FlowMonitor.xml`. Níže ve výpisu 6.11 je část výstupu vygenerovaného textu ze souboru `*.xml`.

Výpis 6.11: Obsah souboru FlowMonitor.xml

```

1 <?xml version="1.0" ?>
2 <FlowMonitor>
3   <FlowStats>
4     <Flow flowId="1"
5       timeFirstTxPacket="+20214285.0ns"
6       timeFirstRxPacket="+20214417.0ns"
7       timeLastTxPacket="+20214285.0ns"
8       timeLastRxPacket="+20214549.0ns"
9       delaySum="+396.0ns" jitterSum="+132.0ns"
10      lastDelay="+264.0ns"
11      txBytes="328" rxBytes="328"
12      txPackets="2" rxPackets="2"
13      lostPackets="0"
14      timesForwarded="0">
15       <delayHistogram nBins="1" >
16         <bin index="0" start="0" width="0.001" count="2" />
17       </delayHistogram>
18       <jitterHistogram nBins="1" >
19         <bin index="0" start="0" width="0.001" count="1" />
20       </jitterHistogram>
21       <packetSizeHistogram nBins="9" >
22         <bin index="8" start="160" width="20" count="2" />
23       </packetSizeHistogram>
24       <flowInterruptionsHistogram nBins="0" >
25       </flowInterruptionsHistogram>
26     </Flow>

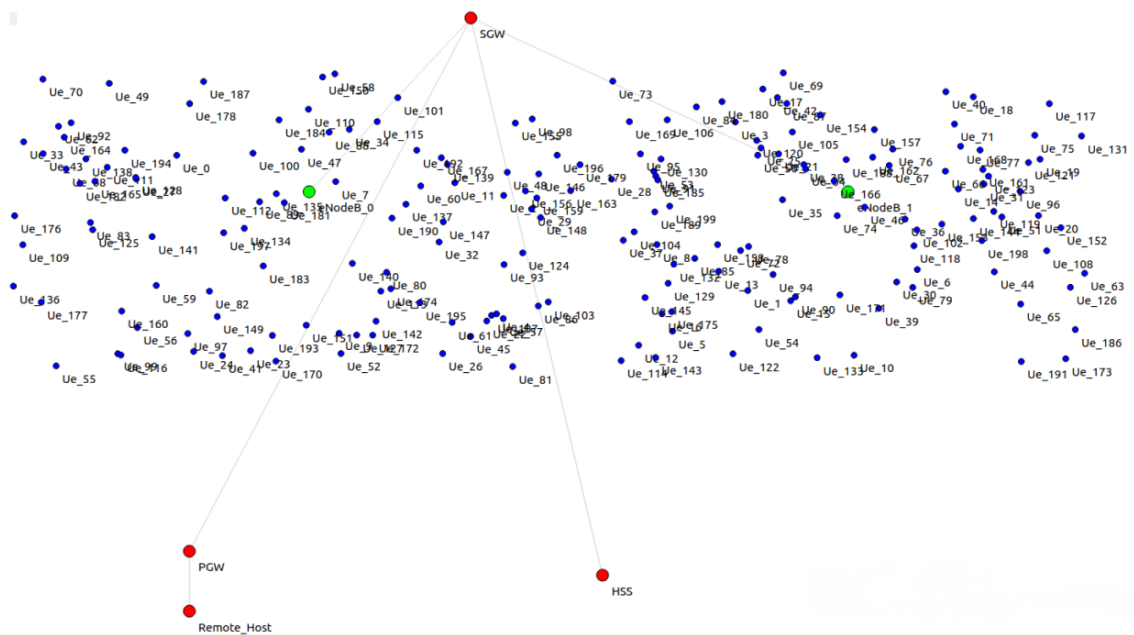
```

Ve výpisu 6.11 je část vygenerovaného textu pomocí metody FlowMonitor, která shromažďuje parametry sítě jako je například celkové zpoždění, jitter nebo zahozené pakety.

6.2 Rozšířený scénář NS-3

V další sekci byly simulovány scénáře, které testovali chování sítě při konkrétních nastaveních pro jednotlivá množství UEs (User Equipments). Oproti počátečnímu scénáři zde bylo několik výrazných změn a to především v množství připojených UEs, změnách v použité aplikaci a v použití nových nástrojů.

Scénáře se zaměřili na technologii Smart Grids, konkrétně na oblast Smart Metering. Je zde využita technologie LTE Cat-M1. Základní topologie scénáře je zobrazena na obrázku 6.2.



Obr. 6.2: Topologie sítě rozšířeného scénáře v programu NetAnim

6.2.1 Výpis kódu

Výpis 6.12: Definice proměnných na začátku

```

1 uint16_t numberOfNodes = 100;
2 uint16_t numberOf_eNodeBs = 2;
3 double simTime = 150;
4 double distance = 90.0;
5
6 bool useCa = false;
7 bool usePcap = false;
8 bool useConfigLTECatM1 = true;

```

Ve výpisu 6.12 je několik parametrů jako `numberOfNodes`, `simTime`, `distance` nebo `useConfigLTECatM1`.

- Proměnná `numberOfNodes` byla zvolena v hodnotách 100, 200 a 300, a to v jednotlivých scénářích.
- Proměnná `simTime` označuje dobu trvání samotné simulace. Tato časová hodnota je udávána v sekundách, ale reálný čas simulace se liší oproti zadané hodnotě, a to v závislosti na množství vykonaných operacích v simulaci.
- Proměnná `distance` udává vzdálenost mezi eNodeBs a je v jednotkách metrů.
- Proměnná `useConfigLTECatM1` je typu `bool`, tedy nabývá hodnot 1 nebo 0. V tomto případě má funkci zapnutí nebo vypnutí pro nastavení konfigurace LTE Cat M1.

Výpis 6.13: Definice argumentů pro příkazový řádek

```

1 CommandLine cmd;
2 cmd.AddValue("numberOfNodes", "Number of eNodeBs + UE pairs",
3     numberOfNodes);
4 cmd.AddValue("simTime", "Total duration of the simulation
5     [s]", simTime);
6 cmd.AddValue("distance", "Distance between eNBs [m]",
7     distance);
8 cmd.AddValue("useCa", "Whether to use carrier aggregation.",
9     useCa);
10 cmd.AddValue("useConfigLTECatM1", "Turn on parameters for
11     LTE Cat-M1", useConfigLTECatM1);
12 cmd.Parse(argc, argv);

```

Kód 6.13 obsahuje definice parametrů, které lze definovat při spuštění simulace přímo z příkazového řádku. Například si můžeme definovat jiný počet UEs než jsou původně definované.

Výpis 6.14: Příkazový řádek

```

1 ./waf --run "scratch/lte-cat-m1 --numberOfNodes=200"

```

To lze provést přímo v příkazovém řádku pomocí příkazu jak je uveden v 6.14. Následně po příkazu dojde ke spuštění simulace a provede se změna v parametru `numberOfNodes`.

Výpis 6.15: Konfigurace vysílacích parametrů

```

1 if (useConfigLTECatM1)
2 {
3     Config::SetDefault("ns3::LteUePhy::TxPower",
4         StringValue("23"));
5     Config::SetDefault("ns3::LteUePhy::NoiseFigure",
6         StringValue("9"));
7     Config::SetDefault("ns3::LteEnbPhy::TxPower",
8         DoubleValue(46.0));
9     Config::SetDefault("ns3::LteEnbPhy::NoiseFigure",
10        StringValue("5"));
11    Config::SetDefault("ns3::LteEnbNetDevice::UlBandwidth",
12        StringValue("6"));
13    Config::SetDefault("ns3::LteEnbNetDevice::DlBandwidth",
14        StringValue("6"));
15 }

```

Pro nastavení sítě, která se podobá technologii LTE Cat-M1, se ve výpise 6.15 nastavily parametry sítě pro UEs a eNodeBs. Vysílací výkon `TxPower` je nastaven pro UEs na hodnotu 23 dBm a pro eNodeBs na 46 dBm. Hodnota šumu `NoiseFigure` je nastavena pro UEs na hodnotu 9 a pro eNodeBs na hodnotu 5. Šířka pásma (`UlBandwidth` a `DlBandwidth`) je nastavena podle hodnoty PRB (Physical Resource Block – Blok Fyzických Zdrojů) na hodnotu 6 [22].

Výpis 6.16: Konfigurace SRS

```
1 Config::SetDefault("ns3::LteEnbRrc::SrsPeriodicity",
2   IntegerValue(320));
```

Ve výpisu 6.16 se nastavuje hodnota SRS, která má vliv na množství připojených zařízení, a může nabývat hodnot 0, 2, 5, 10, 20, 40, 80, 160 a 320. Jedná se o referenční signál SRS (Sounding Reference signal), který se posílá od UE v uplinku směrem k eNodeB. Za pomoci SRS se dělá odhad kvality kanálu podle kterého se může upravovat scheduling frekvence, časování a řízení výkonu. Referenční signál SRS je nezávislý na konkrétním kanálu. Hlavně z pohledu frekvence v rámci pásma je SRS důležité [25].

Výpis 6.17: Výpis do konzole o délce simulace

```
1 auto start = std::chrono::system_clock::now();
2           :
3 auto end = std::chrono::system_clock::now();
4 std::chrono::duration<double> elapsed_seconds = end-start;
5 std::cout << "Time of the simulation [h]: " <<
6   (elapsed_seconds.count())/60/60 << std::endl;
```

Ve výpisu 6.17 se nachází proměnná `start` a `end`, které měří čas kdy simulace započala a kdy skončila. Samotný rozdíl těchto dvou hodnot je v proměnné `elapsed_seconds` a udává celkový čas simulace v sekundách.

Výpis 6.18: Vytvoření uzlů PGW a Remote_Host

```
1 Ptr<Node> pgw = epcHelper->GetPgwNode();
2
3 NodeContainer remoteHostContainer;
4 remoteHostContainer.Create(1);
5 Ptr<Node> remoteHost = remoteHostContainer.Get(0);
6 InternetStackHelper internet;
7 internet.Install(remoteHostContainer);
```

Vytvoření uzlů PGW a Remote_Host je znázorněno ve výpisu 6.18. Jedná se o dva základní uzly, které jsou na hraně mezi prostředím doménových služeb a EPC.

Výpis 6.19: Nastavení internetu část 1 (linka)

```

1 PointToPointHelper p2ph;
2 p2ph.SetDeviceAttribute("DataRate", DataRateValue(
3     DataRate("100Gb/s")));
4 p2ph.SetDeviceAttribute("Mtu", UIntegerValue(1500));
5 p2ph.SetChannelAttribute("Delay", TimeValue(Seconds(0.010)));
6
7 NetDeviceContainer internetDevices =
8     p2ph.Install(pgw, remoteHost);

```

Ve výpisu 6.19 se vytvoří linka mezi Remote_Host a PGW, které jsou nastaveny parametry jako například DataRate (Přenosová rychlost), Mtu (Maximum transmission unit - maximální velikosti IP datagramu) nebo Delay (Zpoždění). Je to hlavní brána mezi internetem a sítí LTE Cat-M1.

Výpis 6.20: Nastavení internetu část 2 (IP adresy)

```

1 Ipv4AddressHelper ipv4h;
2 ipv4h.SetBase("1.0.0.0", "255.0.0.0");
3 Ipv4InterfaceContainer internetIpIfaces = ipv4h.
4     Assign(internetDevices);
5
6 Ipv4Address remoteHostAddr = internetIpIfaces.GetAddress(1);
7
8 Ipv4StaticRoutingHelper ipv4RoutingHelper;
9 Ptr<Ipv4StaticRouting> remoteHostStaticRouting =
10 remoteHostStaticRouting->AddNetworkRouteTo(
11     Ipv4Address("7.0.0.0"), Ipv4Mask("255.0.0.0"), 1);

```

Výpis 6.20 zobrazuje nastavení sítě z hlediska IP adres. Pro UEs je použit rozsah 1.0.0.x a pro cílovou adresu 7.0.0.x. Pro příkaz internetIpIfaces.GetAddress(1), jednička představuje rozhraní p2p (point-to-point) a nula představuje localhost.

Výpis 6.21: Přiřazení IP adres k UEs

```

1 for (uint32_t u = 0; u < ueNodes.GetN(); ++u)
2 {
3     Ptr<Node> ueNode = ueNodes.Get(u);
4
5     Ptr<Ipv4StaticRouting> ueStaticRouting = ipv4RoutingHelper.
6         GetStaticRouting(ueNode->GetObject<Ipv4>());
7     ueStaticRouting->SetDefaultRoute(epcHelper->
8         GetUeDefaultGatewayAddress(), 1);
9 }

```

Přiřazení IP adres k jednotlivým UEs je provedeno ve výpisu 6.21. Je využito statické směrování a nastavení výchozí brány na adresu SGW.

Výpis 6.22: Vytvoření UE_n a eNodeB_n

```
1 NodeContainer ueNodes;  
2 NodeContainer enbNodes;  
3 enbNodes.Create(numberOf_eNodeBs);  
4 ueNodes.Create(numberOfNodes);
```

Vytvoření požadovaného počtu UEs a eNodeB je znázorněno ve výpisu 6.22.

Výpis 6.23: Mobility model

```
1 MobilityHelper mobility;  
2 MobilityHelper enbMobility;  
3 enbMobility.SetMobilityModel(  
4     "ns3::ConstantPositionMobilityModel");  
5 mobility.SetMobilityModel(  
6     "ns3::ConstantPositionMobilityModel");  
7  
8 mobility.SetPositionAllocator (  
9     "ns3::RandomBoxPositionAllocator",  
10    "X", StringValue ("ns3::UniformRandomVariable  
11        [Min=0|Max=180]"),  
12    "Y", StringValue ("ns3::UniformRandomVariable  
13        [Min=10|Max=60]"),  
14    "Z", StringValue ("ns3::UniformRandomVariable  
15        [Min=0|Max=0]"));  
16  
17 enbMobility.Install(enbNodes);  
18 mobility.Install(ueNodes);
```

Ve výpisu 6.23 je nastavena mobilita všech uzlů, které jsou definované v simulaci. Mobilita je rozdělena na mobilitu pro UEs a eNodeB.

- Pro UEs jsou nastaveny na konstantní pozici a během simulace se nemění.
- Pro eNodeBs jsou nastaveny na konstantní pozici, ale jejich rozmístění je dáno náhodně v konkrétní oblasti. Tato oblast je dána osou X a Y.

Výpis 6.24: Připojení UE na eNodeB v aktuálním čase

```
1 double TimeToConnectUE = 0;  
2  
3 for (uint16_t i = 0; i < numberOfNodes; i++)  
4 {  
5     //Sudá ID čísla UEs
```



```

6   if(!(i%2))
7   {
8       Simulator::Schedule(Seconds(TimeToConnectUE), &AttachUE,
9           ueLteDevs.Get(i), enbLteDevs.Get(0), lteHelper);
10  }
11  //Lichá ID čísla UEs
12  else
13  {
14      Simulator::Schedule(Seconds(TimeToConnectUE), &AttachUE,
15          ueLteDevs.Get(i), enbLteDevs.Get(1), lteHelper);
16  }
17
18  TimeToConnectUE = TimeToConnectUE + 0.35;
19 }

```

Samotné připojování UEs k eNodeBs je provedeno ve výpisu 6.24. Pomocí funkce `Simulator::Schedule()` se využilo postupného připojování UEs k eNodeBs v závislosti na čase. Pomocí proměnné `TimeToConnectUE` se nastavuje čas připojení. V závislosti na zvyšující se hodnotě ID UEs se s každým opakujícím cyklem zvýší proměnná o 0,35 sekund.

Výpis 6.25: Vytvoření BulkSendApplikace

```

1  uint16_t port = 9;
2  uint16_t port2 = 33;
3
4  BulkSendHelper source ("ns3::TcpSocketFactory",
5      InetSocketAddress (remoteHostAddr, port));
6  BulkSendHelper source2 ("ns3::TcpSocketFactory",
7      InetSocketAddress (remoteHostAddr, port2));
8
9  source.SetAttribute ("MaxBytes", UintegerValue (3000000));
10 source2.SetAttribute ("MaxBytes", UintegerValue (3000000));
11
12 for (uint16_t i = 0; i < numberOfNodes; i++)
13 {
14     if (i < 2) {
15         source.Install (ueNodes.Get(i));
16     } else {
17         source2.Install (ueNodes.Get(i));
18     }
19 }

```

Aplikace, která běží v síťovém prostředí je uvedena ve výpisu 6.25. Jedná se o datový provoz, který je odeslán z UEs a je proveden pomocí TCP paketů. Hodnota `MaxBytes` udává velikost přenesených dat v jeden okamžik. Protože se bavíme o využití Smart Meteringu, kde provoz může nabývat hodnot cca 2-5 MB za den, proto je zvolen jednorázový datový objem 3 MB.

Výpis 6.26: Vytvoření PacketSinkAplikace

```

1 PacketSinkHelper sink ("ns3::TcpSocketFactory",
2     InetSocketAddress (Ipv4Address::GetAny (), port));
3 ApplicationContainer sinkApps = sink.Install (remoteHost);
4 sinkApps.Start (Seconds (0.0));
5 sinkApps.Stop (Seconds (100));
6
7 PacketSinkHelper sink2 ("ns3::TcpSocketFactory",
8     InetSocketAddress (Ipv4Address::GetAny (), port2));
9 ApplicationContainer sinkApps2 = sink2.Install (remoteHost);
10 sinkApps2.Start (Seconds (0.0));
11 sinkApps2.Stop (Seconds (100));
12
13 lteHelper->EnableTraces();

```

Ve výpisu 6.26 je vytvořena PacketSinkAplikace, která slouží pro příjem na straně serveru.

Výpis 6.27: Definice animace část 1

```

1 AnimationInterface::SetConstantPosition(remoteHost, 30, 100);
2 AnimationInterface::SetConstantPosition(pgw, 30, 90);
3 AnimationInterface anim("1_lte-full.xml");
4 anim.EnablePacketMetadata(true);
5
6 anim.SetMaxPktsPerTraceFile(testValue);
7 anim.UpdateNodeDescription(pgw, "PGW");
8 anim.UpdateNodeDescription(remoteHost, "Remote_Host");
9 anim.UpdateNodeDescription (1, "SGW");
10 anim.UpdateNodeDescription (2, "HSS");
11
12 for (uint32_t u = 0; u < 4; u++)
13 {
14     anim.UpdateNodeSize (u, 2, 2);
15 }

```

Výpis 6.27 obsahuje nastavení uzlů, které se týkají uzlů `Remote_Host`, `PGW`, `SGW` a `HSS`. Dochází k nastavení konstantních pozic a názvů uzlů. Dále je zde

definován soubor pro uložení vizuální simulace a zapnutí metadat. For cyklus slouží pro nastavení velikosti uzlu v animaci, který je určen pro první čtyři uzly.

Výpis 6.28: Definice UEs a animace část 2

```
1 for (uint32_t u = 0; u < ueNodes.GetN(); ++u)
2 {
3     anim.UpdateNodeDescription(ueNodes.Get(u),
4         "Ue_" + std::to_string(u));
5     anim.UpdateNodeColor (ueNodes.Get(u), 0, 0, 255);
6 }
7
8 for (uint32_t u = 0; u < enbNodes.GetN(); ++u)
9 {
10    anim.UpdateNodeDescription(enbNodes.Get(u),
11        "eNodeB_" + std::to_string(u));
12    anim.UpdateNodeColor (enbNodes.Get(u), 0, 255, 0);
13    AnimationInterface::SetConstantPosition(enbNodes.Get(u),
14        50 + distance * u, 30);
15    anim.UpdateNodeSize (u+4, 2, 2);
16 }
```

Nastavení vizuálních prvků pro UEs a eNodeBs je ve výpisu 6.28. Pro UEs a eNodeBs je nastaveno

- očíslování jednotlivých UEs a eNodeBs v závislosti na celkovém počtu uzlů v každé kategorii zvlášť,
- nastavení optimální barvy pro uzly,
- pro eNodeBs nastaveny konstantní pozice vzdálené v jedné linii od sebe vzhledem k parametru `distance`
- a nastavení velikosti uzlu pro eNodeBs.

Výpis 6.29: Nástroj FlowMonitor

```
1 FlowMonitorHelper flowMonHelper;
2 Ptr<FlowMonitor> monitor;
3 monitor = flowMonHelper.Install(enbNodes);
4 monitor = flowMonHelper.Install(ueNodes);
5 monitor = flowMonHelper.Install(remoteHost);
6                                     :
7 monitor->CheckForLostPackets();
8 Ptr<Ipv4FlowClassifier> classifier =
9     DynamicCast<Ipv4FlowClassifier>(
10         flowMonHelper.GetClassifier());
11 std::map<FlowId, FlowMonitor::FlowStats> stats =
```

```

12     monitor->GetFlowStats();
13                                     :
14     monitor->SerializeToXmlFile(
15         "1_manetrouting.flowmon", true, true);

```

Nástroj, který se použije pro získání dat z simulace je uveden ve výpisu 6.29. Senzory, které snímají parametry na jednotlivých uzlech shromažďují data do souboru `1_manetrouting.flowmon`. Tyto informace můžou být volány a dále zpracovány. Jedná se o parametry sítě jako například zpoždění, jitter, ztrátovost paketů, TX bytes, RX bytes, TX packets atd.

Pro vysvětlení pojmů RX/TX bytes a RX/TX packets je vytvořena tabulka 6.1.

Tab. 6.1: Pojmy RX/TX bytes a RX/TX packets

Název	Popis
TX bytes	celkový počet bajtů přenesených přes rozhraní uzlu
RX bytes	celkový počet bajtů přijatých přes rozhraní uzlu
TX packets	celkový počet přenesených paketů v daném uzlu
RX packets	celkový počet přijatých paketů v daném uzlu

Výpis 6.30: Nástroj Gnuplot

```

1  std::string jmenoSouboru = "1_delay";
2  std::string graphicsFileName = jmenoSouboru + ".png";
3  std::string plotFileName = jmenoSouboru + ".plt";
4  std::string plotTitle = "Average delay";
5  std::string dataTitle = "Delay [ms]";
6  Gnuplot gnuplot (graphicsFileName);
7  gnuplot.SetTitle (plotTitle);
8  gnuplot.SetTerminal ("png");
9  gnuplot.SetLegend ("IDs of data streams", "Delay [ms]");
10 gnuplot.AppendExtra("set xrange
11     [1: "+std::to_string(numberOfNodes*2)+" ]");
12 gnuplot.AppendExtra("set yrange [0:7000]");
13 gnuplot.AppendExtra("set grid");
14 Gnuplot2dDataset dataset;
15                                     :
16 double Delay;
17 Delay = (i->second.delaySum.GetSeconds())/
18     i->second.rxPackets)*1000;
19 dataset.Add ((double) i->first ,( double ) Delay ) ;

```

```

20         :
21 gnuplot.AddDataset (dataset);
22 std::ofstream plotFile (plotFileName.c_str());
23 gnuplot.GenerateOutput (plotFile);
24 plotFile.close ();

```

Ve výpisu 6.30 se nachází vytvoření souborů `1_delay.plt`, který dostal název pomocí parametru `plotFileName`. Při vytvoření byl použit nástroj Gnuplot, který slouží pro vygenerování grafů. V samotném souboru s příponou `*.plt` jsou definované následující parametry:

- výstupní formát souboru – `gnuplot.SetTerminal()` (v našem případě se jedná o formát PNG),
- výstupní název souboru PNG – `graphicsFileName`,
- hlavní název v grafu – `plotTitle`,
- název osy X a Y – `gnuplot.SetLegend(X,Y)`,
- měřítko osy X a Y – `gnuplot.AppendExtra()`
- a výsledky z měření – `dataset.Add()`.

Pro všechny ostatní vygenerované grafy je kód stejný pouze je rozdíl v zapsaných informacích především v názvech a ve vypočtení měřeného parametru. Nástroj Gnuplot byl využit i pro získání grafu u přenosové rychlosti a ztrátovosti paketů.

Výpis 6.31: Výpis parametrů do konzole a souboru

```

1  ofstream consoleTXT("1_console.txt");
2  std::cout << std::endl << "*** Flow monitor statistic ***"
3  for (std::map<FlowId, FlowMonitor::FlowStats>::
4      const_iterator i = stats.begin(); i != stats.end(); ++i)
5  {
6      :
7      Ipv4FlowClassifier::FiveTuple t = classifier->
8          FindFlow(i->first);
9      std::cout << " Flow ID:" << i->first << std::endl;
10     consoleTXT << " Flow ID:" << i->first << std::endl;
11     std::cout << " Src add: " << t.sourceAddress <<
12         " -> Dst add: " << t.destinationAddress << std::endl;
13     consoleTXT << " Src add: " << t.sourceAddress <<
14         " -> Dst add: " << t.destinationAddress << std::endl;
15     std::cout << " Src port: " << t.sourcePort <<
16         " -> Dst port: " << t.destinationPort << std::endl;
17     consoleTXT << " Src port: " << t.sourcePort <<
18         " -> Dst port: " << t.destinationPort << std::endl;
19     std::cout << " Throughput: " << i->second.rxBytes * 8.0/

```

```

20         (i->second.timeLastRxPacket.GetSeconds()- i->
21             second.timeFirstTxPacket.GetSeconds())/ 1024 <<
22             " kb/s"<< std::endl;
23     consoleTXT << " Throughput: " << i->second.rxBytes * 8.0/
24         (i->second.timeLastRxPacket.GetSeconds()- i->
25             second.timeFirstTxPacket.GetSeconds())/ 1024 <<
26             " kb/s"<< std::endl;
27     std::cout << " Average delay:      " <<
28         (i->second.delaySum.GetSeconds()/ i->
29             second.rxPackets)*1000 << " ms"<< std::endl;
30     consoleTXT << " Average delay:      " <<
31         (i->second.delaySum.GetSeconds()/ i->
32             second.rxPackets)*1000 << " ms"<< std::endl;
33         :
34     std::cout << " Packet loss:      " << (((i->second.txPackets -
35         i->second.rxPackets)*1.0)/i->second.txPackets)*100 <<
36         " %" << std::endl;
37     consoleTXT << " Packet loss:      " << (((i->second.txPackets -
38         i->second.rxPackets)*1.0)/i->second.txPackets)*100 <<
39         " %" << std::endl;
40         :
41     std::cout << " -----" << std::endl;
42     consoleTXT << " -----" << std::endl;
43 }
44 consoleTXT.close();

```

Ve výpisu 6.31 dochází k zápisu ID toku, zdrojové adresy, cílové adresy, zdrojového portu, cílového portu, přenosové rychlosti, zpoždění a ztrátovosti paketů do konzole a souboru 1_console.txt. Zápis informací do konzole se provede vždy po skončení simulace. Samotný přepis informací do souboru je proveden za pomoci příkazu ofstream <název>("<název_souboru>.txt") a ukončen příkazem <název>.close().

Ztrátovost paketů je vyjádřena v procentech jako

$$\alpha = \frac{A - B}{A} \cdot 100,$$

kde α je ztrátovost paketů, A je celkový počet přenesených paketů a B je celkový počet přijatých paketů.

Zpoždění je vyjádřeno v milisekundách jako

$$\beta = \frac{A}{B} \cdot 1\,000$$

kde β je průměrné zpoždění, A je celkové zpoždění a B je celkový počet přijatých paketů.

Přenosová rychlost je vyjádřena v kb/s jako

$$\gamma = \frac{A \cdot 8}{(B - C) \cdot 1024}$$

kde γ je přenosová rychlost, A je celkový počet přijatých bajtů, B je čas posledního přijatého paketu a C je čas prvního přeneseného paketu.

7 Výsledky simulací pro rozšířený scénář

Tato kapitola popisuje výsledky simulovaných scénářů a vychází z vytvořených rozšířených scénářů. Celkem byly vytvořeny 3 scénáře, které se zaměřovali na chování QoS parametrů sítě při různém počtu UEs. První scénář pracoval se 100 UEs, druhý scénář s 200 UEs a třetí s 250 UEs. Maximální hodnota UEs, která byla možná nastavit, byla hodnota 250 UEs. Ve všech scénářích se sledovaly QoS parametry jako přenosová rychlost, zpoždění a ztrátovost paketů.

7.1 Scénář se 100 UEs

V prvním scénáři bylo nastaveno množství UEs na hodnotu 100. Hlavní parametry použité simulace jsou znázorněny ve výpisu 6.15. Níže je přehledná tabulka 7.1 všech parametrů. Simulace scénáře všech částí trvala 2 hodiny a 8 minut.

Tab. 7.1: Přehled parametrů pro scénář se 100 UEs

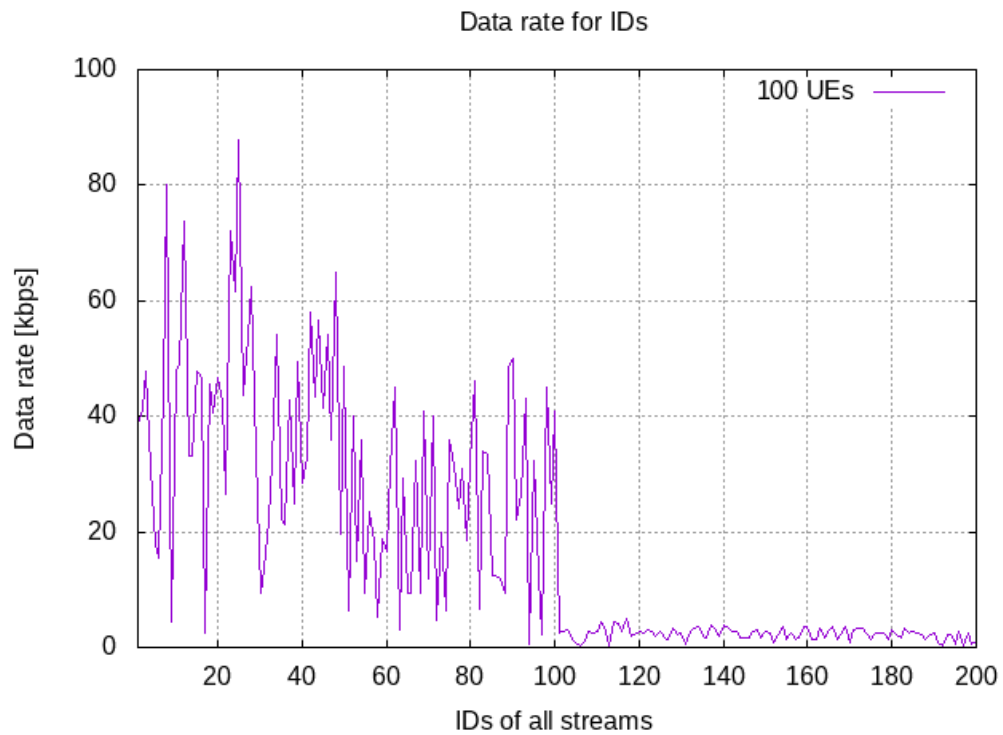
Název	Hodnota
Počet UEs	100
Počet eNodeBs	2
Přenesený vysílací výkon UE	23 dBm
Hodnota šumu UE	9
Přenesený vysílací výkon eNodeB	46 dBm
Hodnota šumu eNodeB	5
Počet PRB pro UL a DL	6
Šířka pásma	1,08 MHz
Hodnota signálu SRS	320 ms

Na obrázku 7.1 je zachycena přenosová rychlost pro jednotlivé ID toky. Přenosová rychlost je udávána v kb/s. Datový provoz směrem od UEs k eNodeBs je znázorněn pro ID toky (1;100) a pro opačný směr provozu je znázorněn pro toky (101;200).

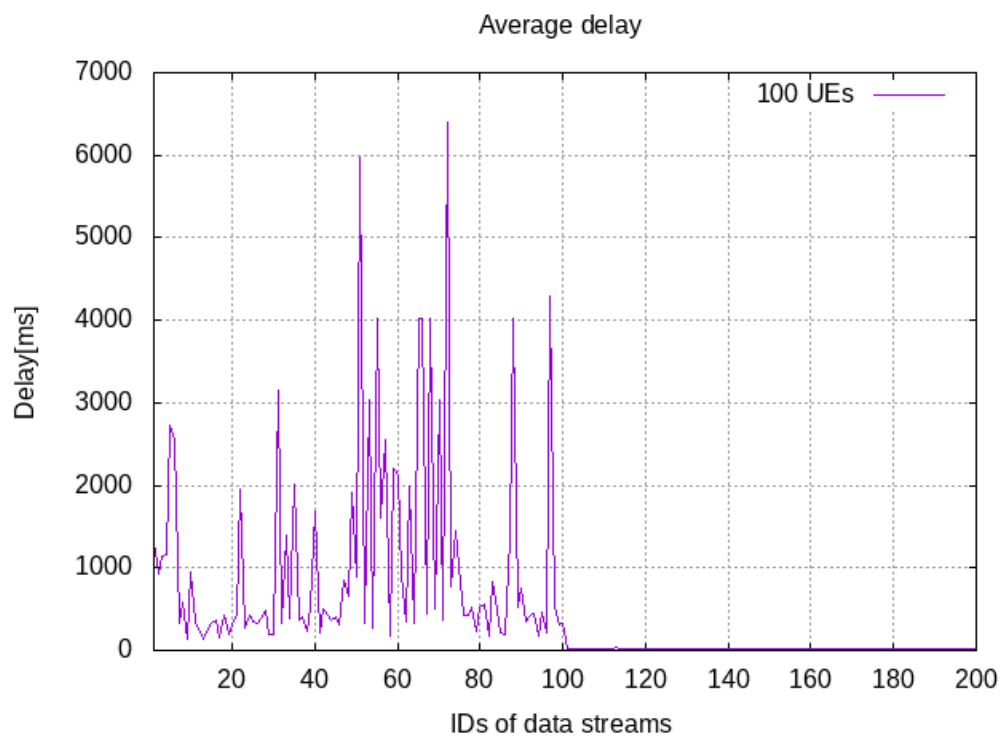
Pro provoz směrem k eNodeB dosahuje maximální přenosové rychlosti 88 kb/s a průměrná hodnota je stanovena na 32,2 kb/s. Pohybujeme se v rozmezí hodnot mezi 10 - 88 kb/s. Pro provoz směrem k UEs se hodnoty pohybují mezi 2 - 12 kb/s.

Na obrázku 7.2 je zobrazena charakteristika zpoždění v síti. Směr datového provozu je opět stejný jako u předešlého grafu.

Pro provoz směrem k eNodeB dosahuje síť maximálního zpoždění 6 405 ms a minimální hodnotu 127 ms. Průměrná hodnota zpoždění směrem k eNodeB odpovídá hodnotě 1 071 ms. Pro provoz směrem k UEs se hodnota zpoždění pohybuje téměř okolo nulových hodnot.



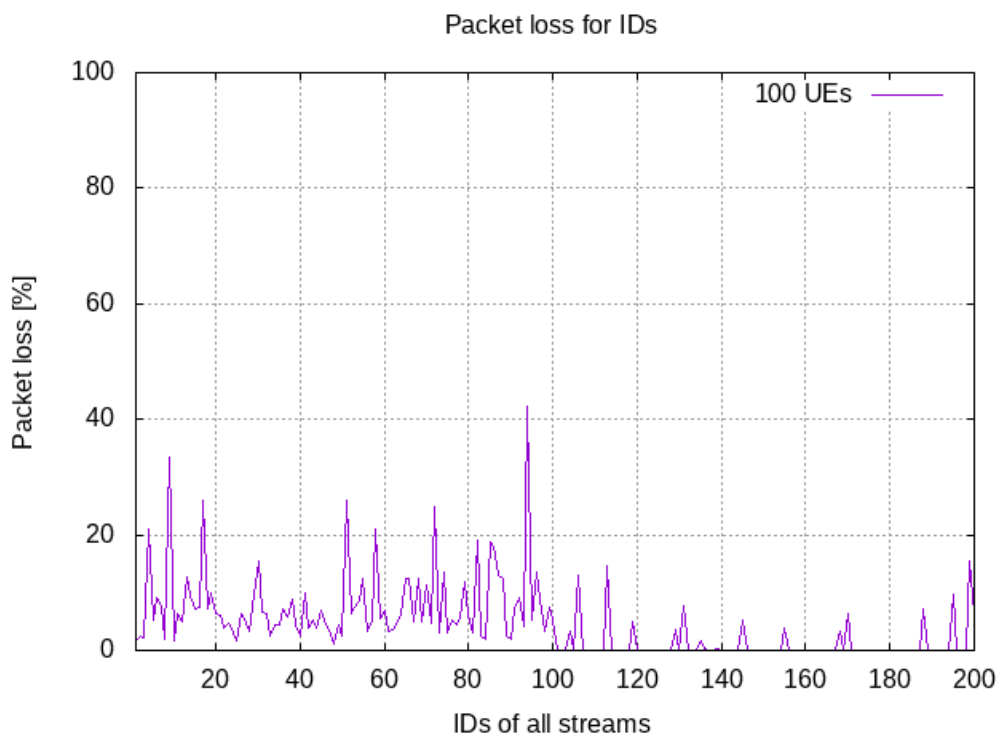
Obr. 7.1: První scénář – Přenosová rychlost



Obr. 7.2: První scénář – Zpoždění

Na obrázku 7.3 je možné vidět ztrátovost paketů napříč jednotlivými uzly. Směr datového provozu je opět stejný jako u předešlého grafu.

Pro provoz směrem k eNodeB byla naměřena největší hodnota ztrátovosti 42 %. Průměrná hodnota směrem k eNodeB činila hodnotu 7,7 %.



Obr. 7.3: První scénář – Ztrátovost paketů

7.2 Scénář s 200 UEs

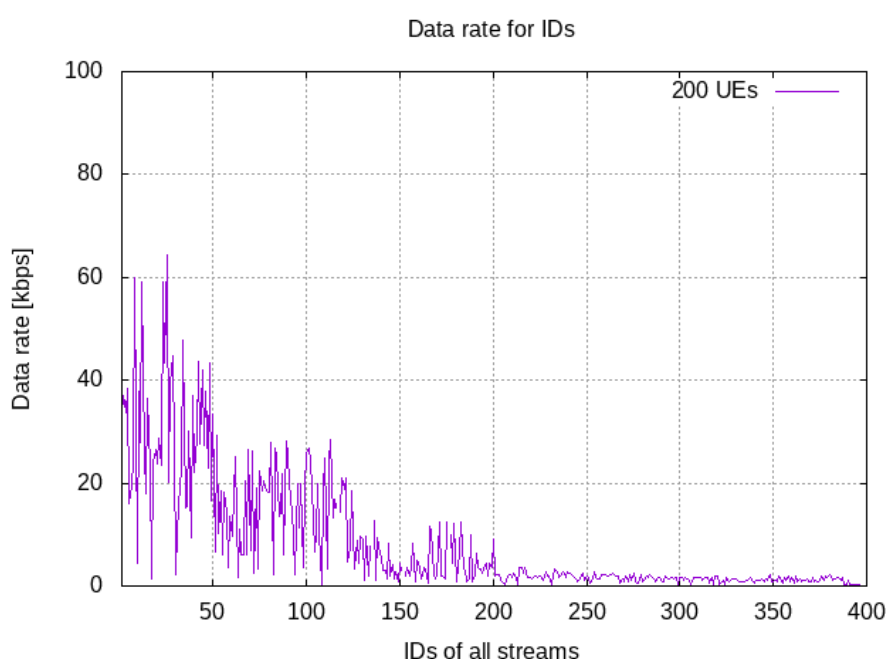
V druhém scénáři bylo nastaveno množství UEs na hodnotu 200. Hlavní parametry použité simulace jsou stejně definované jako pro 100 UEs v podkapitole 7.1. Níže je přehledná tabulka 7.2 všech parametrů. Simulace scénáře všech částí trvala 4 hodiny a 1 minutu.

Na obrázku 7.4 je zachycena přenosová rychlost pro jednotlivé ID toky. Datový provoz směrem od UEs k eNodeBs je znázorněn pro ID toky (1;200) a pro opačný směr provozu je znázorněn pro toky (201;400).

Pro provoz směrem k eNodeB dosahuje maximální přenosové rychlosti 64 kb/s a průměrná hodnota je stanovena na 15 kb/s. Pohybujeme se v rozmezí hodnot mezi 2 - 64 kb/s. Pro provoz směrem k UEs se hodnoty pohybují mezi 0,1 - 4 kb/s.

Tab. 7.2: Přehled parametrů pro scénář s 200 UEs

Název	Hodnota
Počet UEs	200
Počet eNodeBs	2
Přenesený vysílací výkon UE	23 dBm
Hodnota šumu UE	9
Přenesený vysílací výkon eNodeB	46 dBm
Hodnota šumu eNodeB	5
Počet PRB pro UL a DL	6
Šířka pásma	1,08 MHz
Hodnota signálu SRS	320 ms



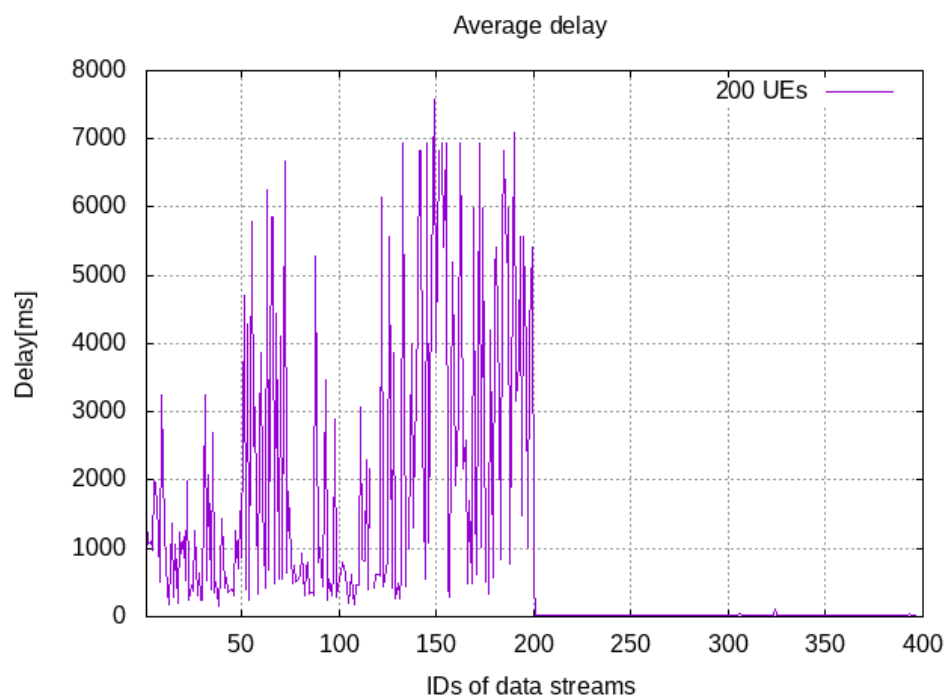
Obr. 7.4: Druhý scénář – Přenosová rychlost

Na obrázku 7.5 je zobrazena charakteristika zpoždění v síti. Směr datového provozu je opět stejný jako u předešlého grafu.

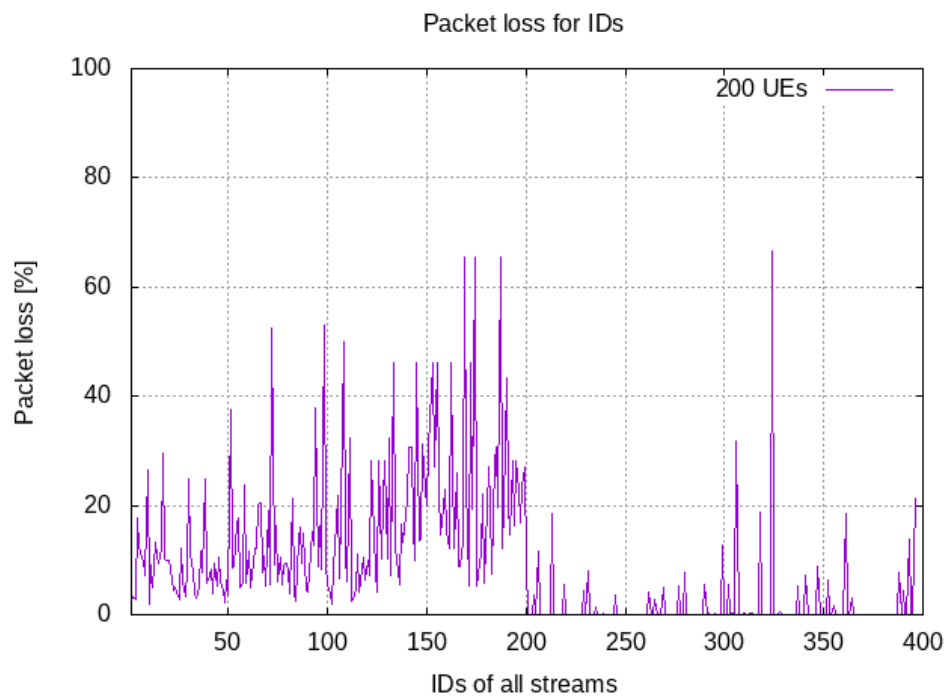
Pro provoz směrem k eNodeB dosahuje síť maximálního zpoždění 7 584 ms a minimální hodnotu 156 ms. Průměrná hodnota zpoždění směrem k eNodeB odpovídá hodnotě 2 119 ms. Pro provoz směrem k UEs se hodnota zpoždění pohybuje téměř okolo nulových hodnot.

Na obrázku 7.6 je možné vidět ztrátovost paketů napříč jednotlivými uzly. Směr datového provozu je opět stejný jako u předešlého grafu.

Pro provoz směrem k eNodeB byla naměřena největší hodnota ztrátovosti 65 %. Průměrná hodnota směrem k eNodeB činila hodnotu 15 %.



Obr. 7.5: Druhý scénář – Zpoždění



Obr. 7.6: Druhý scénář – Ztrátovost paketů

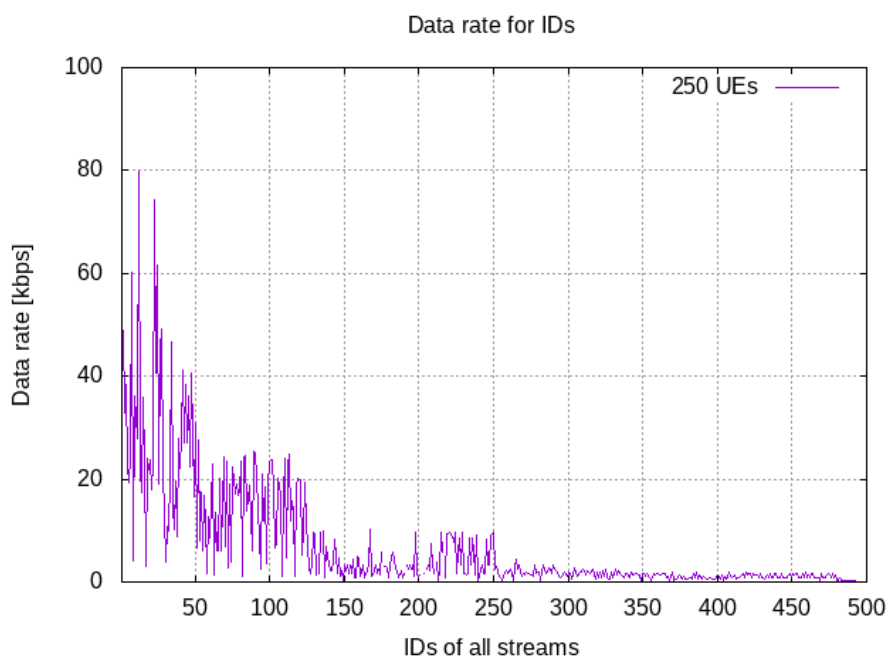
7.3 Scénář s 250 UEs

V třetím scénáři bylo nastaveno množství UEs na hodnotu 250. Hlavní parametry použité simulace jsou stejně definované jako pro 100 UEs v podkapitole 7.1. Níže je přehledná tabulka 7.3 všech parametrů. Simulace scénáře všech částí trvala 5 hodin a 6 minut.

Tab. 7.3: Přehled parametrů pro scénář s 250 UEs

Název	Hodnota
Počet UEs	250
Počet eNodeBs	2
Přenesený vysílací výkon UE	23 dBm
Hodnota šumu UE	9
Přenesený vysílací výkon eNodeB	46 dBm
Hodnota šumu eNodeB	5
Počet PRB pro UL a DL	6
Šířka pásma	1,08 MHz
Hodnota signálu SRS	320 ms

Na obrázku 7.7 je zachycena přenosová rychlost pro jednotlivé ID toky. Datový provoz směrem od UEs k eNodeBs je znázorněn pro ID toky (1;250) a pro opačný směr provozu je znázorněn pro toky (251;500).

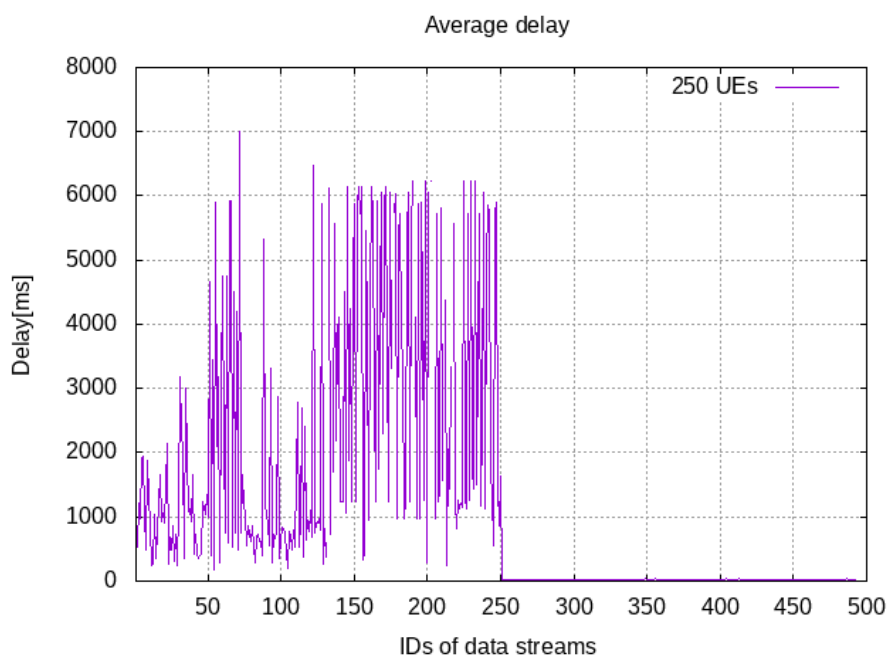


Obr. 7.7: Třetí scénář – Přenosová rychlost

Pro provoz směrem k eNodeB dosahuje maximální přenosové rychlosti 80 kb/s a průměrná hodnota je stanovena na 12,5 kb/s. Pohybujeme se v rozmezí hodnot mezi 2 - 80 kb/s. Pro provoz směrem k UEs se hodnoty pohybují mezi 0,1 - 4,5 kb/s.

Na obrázku 7.8 je zobrazena charakteristika zpoždění v síti. Směr datového provozu je opět stejný jako u předešlého grafu.

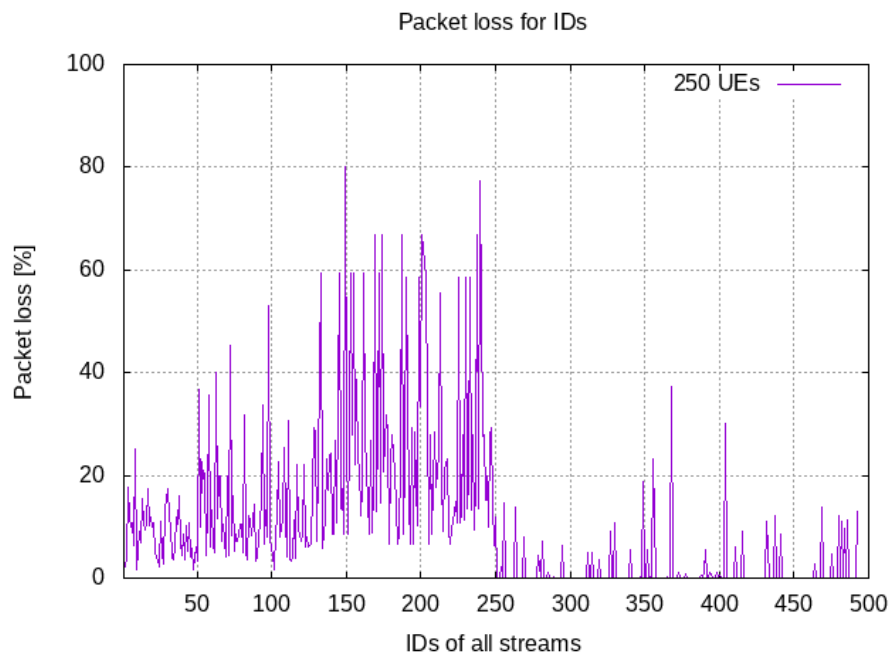
Pro provoz směrem k eNodeB dosahuje síť maximálního zpoždění 7 003 ms a minimální hodnotu 184 ms. Průměrná hodnota zpoždění směrem k eNodeB odpovídá hodnotě 2 349 ms. Pro provoz směrem k UEs se hodnota zpoždění pohybuje téměř okolo nulových hodnot.



Obr. 7.8: Třetí scénář – Zpoždění

Na obrázku 7.9 je možné vidět ztrátovost paketů napříč jednotlivými uzly. Směr datového provozu je opět stejný jako u předešlého grafu.

Pro provoz směrem k eNodeB byla naměřena největší hodnota ztrátovosti 80 %. Průměrná hodnota směrem k eNodeB činila hodnotu 18 %.



Obr. 7.9: Třetí scénář – Ztrátovost paketů

7.4 Přehled výsledků v tabulce

Přehled výsledků jednotlivých rozšířených scénářů je v tabulce 7.4. Najdeme zde výsledky z QoS parametrů jako přenosová rychlost, zpoždění a ztrátovost paketů.

Tab. 7.4: Přehled výsledků pro uplink

Název/Počet UEs	100 UEs	200 UEs	250 UEs
Čas celkové simulace	2 h 8 min	4 h 1 min	5 h 6 min
Maximální přenosová rychlost	88 kb/s	64 kb/s	80 kb/s
Průměrná přenosová rychlost	32,2 kb/s	15 kb/s	12,5 kb/s
Maximální hodnota zpoždění	6 405 ms	7 584 ms	7 003 ms
Minimální hodnota zpoždění	127 ms	156 ms	184 ms
Průměrná hodnota zpoždění	1 071 ms	2 199 ms	2 349 ms
Maximální ztrátovost	42 %	65 %	80 %
Průměrná ztrátovost	7,7 %	15 %	18 %

Z výsledků lze usoudit, že při vyšším počtu UEs dochází k větším nárokům na síť. Zejména došlo k

- vyššímu nárůstu ztrátovosti paketů
- a k vyšším hodnotám zpoždění.

Tím je způsobena nižší rychlost přenosu a nižší celková kvalita služeb.

Hodnoty maximální přenosové rychlosti jsou o něco větší než jsou uvedené v literatuře [10] z měření, kde je zmíněna hodnota 52 kb/s. Ve výstupech bakalářské práce se hodnoty pohybují v rozmezí 64 - 88 kb/s.

V literatuře [27] je průměrné zpoždění LTE Cat-M1 u protokolu TCP pouze 2,86 sekund. V porovnání s výsledky simulací v bakalářské práci, kde se hodnoty pohybují v rozmezí 1,071 - 2,349 sekund, jsou dosažené hodnoty lepší než ve zmiňované literatuře. Můžeme tedy říct, že technologie LTE Cat-M1 splňuje požadavky pro aplikace SmartGrids z hlediska zpoždění.

Ztrátovost paketů by se měla pohybovat okolo 0 % při TCP protokolu, ale v našem případě to neplatí. V literatuře [27] je hodnota ztrátovosti při UDP protokol okolo 5,8 až 6,7 % a i v tomto případě je ztrátovost větší. Z toho lze vyvodit závěr, že vhodnost naší simulace v oblasti SmartGrids je pouze pro aplikace, které si můžou dovolit ztratit menší množství dat například pro GPS aplikace, kde menší výpadek nevadí.

Spolehlivost sítě byla díky použití TCP protokolu větší než by bylo s UDP protokolem, ale mělo to větší vliv na vytížení sítě.

Použití protokolu TCP může sice zajistit spolehlivé doručení dat přes síť LTE Cat-M1. Nicméně však vysoká míra ztráty paketů a dlouhé zpoždění paketů v sítích LTE Cat-M1 mohou snadno způsobit vypršení časového limitu připojení TCP na dlouhou dobu (tj. zastavení přenosu) a mohou dokonce způsobit ukončení připojení TCP. Ze zmiňovaného důvodu většina IoT aplikací může upřednostňovat k odesílání paketů protokol UDP namísto TCP [27]. V této bakalářské práci byla měřena síť LTE Cat-M1, kde koncová zařízení používala protokol TCP, a to má vliv na hodnoty zpoždění a ztrátovosti paketů.

Problém, který mohl způsobit větší hodnoty u zpoždění a ztrátovosti paketů, mohl vzniknout při připojování UEs k eNodeB. Pro další práci by bylo vhodné nastavit k funkci připojení, tak i odpojení UEs po provedení odeslání dat. Tento jev může mít vliv na menší vytížení sítě.

Závěr

Bakalářská práce se zabývala problematikou dvou licenčních technologií LTE Cat-M1 a NB-IoT, které patří mezi LPWAN sítě.

V první části semestrální práce jsou popsány obecně LPWAN sítě a uplatnění LPWAN v rámci SmartGrids. Dále jsou popsány SmartGrids, které v dnešní době mají velký potenciál stát se přínosem pro každodenní život. V poslední sekci první části je popsáno samotné uplatnění v LPWAN sítí ve SmartGrids.

Druhá část se věnuje popisu NB-IoT a rozebírá v podrobnostech jeho technické parametry.

Třetí část pojednává o další z LPWAN technologií, tentokrát LTE Cat-M1. Popisuje se zde architektura LTE sítě, systémový model, technické parametry a pokrytí. Část pokrytí se zaměřuje především na vylepšení toho parametru.

Čtvrtá část komplexně srovnává obě popisované technologie po technické i aplikační stránce. Srovnání popisuje spotřebu, pokrytí, globální dostupnost, rozsah pokrytí, připojení a cenu.

Pátá část se soustředí na síťový simulátor NS-3, jeho popis a modul LENA. Navazující část popisuje vytvořený počáteční scénář, kde je implementována technologie LTE s modulem LENA. Byl vytvořen ukázkový scénář, kde probíhala síťová komunikace pomocí UDP paketů pro ověření funkčnosti a principů simulátoru. Dosažené výsledky pomohli v bakalářské práci navázat komplexnějšími simulacemi se zaměřením do oblasti SmartGrids.

Šestá část se zaměřuje na rozšířený scénář, kde jsou využity vlastnosti technologie LTE Cat-M1 a poznatky z oblasti SmartGrids. Došlo k popisu jednotlivých scénářů, kde celkem byly vytvořeny tři scénáře. Výpisy výsledků simulací byly rozděleny do tří podkapitol, které byly roztrženy podle počtu UEs na 100, 200 a 250. Výsledky jednotlivých scénářů lze najít v kapitole 7. V každé části jsou popsány grafy a jejich hodnoty pro přenosovou rychlost, zpoždění a ztrátovost paketů. V poslední podkapitole 7.4 došlo k porovnání a diskutování dosažených výsledků v bakalářské práci.

Pro možnost budoucího pokračování v práci je nutné překonat problém s hranicí maximálního počtu UEs, který je možné nastavit. Dále potlačit velikosti hodnot u ztrátovosti paketů a to optimalizací pomocí chytrého připojení v závislosti na využití sítě.

Literatura

- [1] Beyene, Y. D.; Jantti, R.; Tirkkonen, O.; aj.: NB-IoT Technology Overview and Experience from Cloud-RAN Implementation. *IEEE Wireless Communications*, ročník 24, č. 3, 2017: s. 26–32, ISSN 1536-1284, doi:10.1109/MWC.2017.1600418.
- [2] Chakrapani, A.: NB-IoT Uplink Receiver Design and Performance Study. *IEEE Internet of Things Journal*, ročník 7, č. 3, 2020: s. 2469–2482, doi:10.1109/JIOT.2019.2957641.
- [3] U-blox Introduces Plans to Support Verizon’s New LTE Cat M1 Network for IoT. *Wireless News*, 2016.
URL <https://www.proquest.com/docview/1831197293?accountid=17115>
- [4] Dawaliby, S.; Bradai, A.; Pousset, Y.: In depth performance evaluation of LTE-M for M2M communications. In *2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2016, s. 1–8, doi:10.1109/WiMOB.2016.7763264.
- [5] GSMA: GSMA Mobile IoT LPWA. 2020.
URL <https://www.gsma.com/iot/mobile-iot-commercial-launches/>
- [6] GSMA: Mobile IoT Deployment Map. 2020.
URL <https://www.gsma.com/iot/deployment-map/>
- [7] Hsieh, P.-C.; Jia, Y.; Parra, D.; aj.: An Experimental Study on Coverage Enhancement of LTE Cat-M1 for Machine-Type Communication. In *IEEE International Conference on Communications*, Institute of Electrical and Electronics Engineers Inc, 2018, ISBN 9781538631805, ISSN 15503607, doi:10.1109/ICC.2018.8422888.
- [8] Jawhar, I.; Mohamed, N.; Al-jaroodi, J.: Networking architectures and protocols for smart city systems. *Journal of Internet Services and Applications*, ročník 9, č. 1, 2018: s. 1–16, ISSN 1867-4828, doi:10.1186/s13174-018-0097-0.
- [9] Jiang, N.; Deng, Y.; Condoluci, M.; aj.: RACH Preamble Repetition in NB-IoT Network. *IEEE Communications Letters*, ročník 22, č. 6, 2018: s. 1244–1247, ISSN 1089-7798, doi:10.1109/LCOMM.2018.2793274.
URL <https://ieeexplore-ieee-org.ezproxy.lib.vutbr.cz/document/8258982>

- [10] Kozma, D.; Soás, G.; Ficzer, D.; aj.: Communication Challenges and Solutions between Heterogeneous Industrial IoT Systems. In *2019 15th International Conference on Network and Service Management (CNSM)*, 2019, s. 1–6, doi: 10.23919/CNSM46954.2019.9012664.
- [11] Li, Y.; Cheng, X.; Cao, Y.; aj.: Smart Choice for the Smart Grid. *IEEE Internet of Things Journal*, ročník 5, č. 3, 2018: s. 1505–1515, doi:10.1109/JIOT.2017.2781251.
- [12] Lirette, C.: What are the Differences Between LTE Cat-M1 and NB-IoT Connectivity. 2019.
URL <https://www.soracom.io/blog/>
- [13] Mangalvedhe, N.; Ratasuk, R.; Ghosh, A.: NB-IoT deployment study for low power wide area cellular IoT. *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2016: s. 1–6, ISSN 2166-9589, doi:10.1109/PIMRC.2016.7794567.
URL <http://ieeexplore.ieee.org/document/7794567/>
- [14] Milošević, N.; Nikolić, Z.; Jelenković, F.; aj.: Spectrum Sensing Experimentation for LTE and WiFi Unlicensed Band Operation. *Telfor Journal*, ročník 8, č. 2, 2016: s. 76–80, ISSN 1821-3251, doi:10.5937/telfor1602076M.
URL <https://doaj.org/article/9ec3409ef368426f95cabffd712ac2b4>
- [15] Mroue, H.; Andrieux, G.; Cruz, E. M.; aj.: Evaluation of LPWAN technology for Smart City. *EAI Endorsed Transactions on Smart Cities*, ročník 2, č. 6, 2017, doi:10.4108/eai.20-12-2017.153494.
- [16] NSNAM: Lena-simple-epc.cc File Reference.
URL https://www.nsnam.org/doxygen/lena-simple-epc_8cc.html
- [17] Ogbodo, E. U.; Dorrell, D.; Abu-mahfouz, A. M.: Cognitive Radio Based Sensor Network in Smart Grid. *IEEE Access*, ročník 5, 2017: s. 19084–19098, doi: 10.1109/ACCESS.2017.2749415.
- [18] Persia, S.; Rea, L.: Next generation M2M Cellular Networks. In *2016 AEIT International Annual Conference (AEIT)*, AEIT, 2016, s. 1–6, doi:10.23919/AEIT.2016.7892789.
- [19] Peška, R.: LTE M v otázkách a odpovědích. 2019.
URL <https://vyvoj.hw.cz/lte-m-v-otazkach-a-odpovedich.html>

- [20] Rama, Y.; Ozpmar, M. A.: A Comparison of Long-Range Licensed and Unlicensed LPWAN Technologies According to Their Geolocation Services and Commercial Opportunities. In *Mediterranean Microwave Symposium*, IEEE Computer Society, 2019, ISBN 9781538671320, ISSN 21579822, s. 398–403, doi: 10.1109/MMS.2018.8612009.
- [21] Sabbah, A.; Jarwan, A.; Issa, O.; aj.: Enabling LTE emulation by integrating CORE emulator and LTE-EPC network (LENA) simulator. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, IEEE, 2017, ISBN 9781538635292, s. 1–6, doi: 10.1109/PIMRC.2017.8292642.
- [22] Semiconductor, A.: Coverage Analysis of LTE-M Category-M1. 2017.
URL <https://www.altair-semi.com/wp-content/uploads/2017/02/Coverage-Analysis-of-LTE-CAT-M1-White-Paper.pdf>
- [23] Song, Y.; Lin, J.; Tang, M.; aj.: An Internet of Energy Things Based on Wireless LPWAN. *Engineering (Beijing, China)*, ročník 3, č. 4, 2017: s. 460–466, ISSN 2095-8099, doi:10.1016/J.ENG.2017.04.011.
- [24] Sultan, R.; Refaey, A.; Hamouda, W.: Resource Allocation in CAT-M and LTE-A Coexistence: A Joint Contention Bandwidth Optimization Scheme. In *2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2020, s. 1–6, doi:10.1109/CCECE47787.2020.9255823.
- [25] Tian, H.; Yang, L.; Li, S.: SNR estimation based on sounding reference signal in LTE uplink. In *2013 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC 2013)*, IEEE, 2013, s. 1–5, doi:10.1109/ICSPCC.2013.6664058.
- [26] Wang, A.; Jiang, W.: Research of Teaching on Network Course Based on NS-3. In *2009 First International Workshop on Education Technology and Computer Science*, ročník 2, IEEE, 2009, ISBN 9780769535579, s. 629–632, doi:10.1109/ETCS.2009.401.
- [27] Wang, S.-Y.; Chang, J.-E.; Fan, H.; aj.: Performance Comparisons of NB-IoT, LTE Cat-M1, Sigfox, and LoRa Moving at High Speeds in the Air. In *2020 IEEE Symposium on Computers and Communications (ISCC)*, 2020, s. 1–6, doi:10.1109/ISCC50000.2020.9219557.

A Obsah zdrojového souboru

Simulace byly prováděny za pomoci těchto programů

- VMware Workstation 16 Pro verze 16.0.0
- Network Simulator 3 (NS-3) verze 3.30.1
- NetAnim

```
/ .....kořenový adresář přiloženého USB media
├── LTE_Cat-M1_UE=100 ..... adresář se simulací 100 UE
│   ├── Simulace ..... adresář s výstupními daty
│   │   ├── 1_console.txt
│   │   └── 1_manetrouting.flowmon
│   ├── 1_datarate.plt
│   ├── 1_datarate.png
│   ├── 1_delay.plt
│   ├── 1_delay.png
│   ├── 1_packetloss.plt
│   ├── 1_packetloss.png
│   └── lte-cat-m1-100.cc
├── LTE_Cat-M1_UE=200 ..... adresář se simulací 200 UE
│   ├── Simulace ..... adresář s výstupními daty
│   │   ├── 1_console.txt
│   │   └── 1_manetrouting.flowmon
│   ├── 1_datarate.plt
│   ├── 1_datarate.png
│   ├── 1_delay.plt
│   ├── 1_delay.png
│   ├── 1_packetloss.plt
│   ├── 1_packetloss.png
│   └── lte-cat-m1-200.cc
├── LTE_Cat-M1_UE=250 ..... adresář se simulací 250 UE
│   ├── Simulace ..... adresář s výstupními daty
│   │   ├── 1_console.txt
│   │   └── 1_manetrouting.flowmon
│   ├── 1_datarate.plt
│   ├── 1_datarate.png
│   ├── 1_delay.plt
│   ├── 1_delay.png
│   ├── 1_packetloss.plt
│   ├── 1_packetloss.png
│   └── lte-cat-m1-250.cc
└── Úvodní scénář ..... adresář s počátečním scénářem
    ├── FlowMonitor.xml
    └── lena-simple-epc.cc
```